Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Cybernetics

FACULTY
OF ELECTRICAL
ENGINEERING
CTU IN PRAGUE

# Learning Body Models: From Humans to Humanoids

Habilitation Thesis

**Matěj Hoffmann**

Prague, November 2021

# Acknowledgements

First, I would like to express thanks to my Humanoid and cognitive robotics research group at the Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague. The contributions of Karla Štěpánová, Zdeněk Straka, Petr Švarný, Filipe Gama, Jakub Rozlivek, and Lukáš Rustler constitute an important part of the thesis. Second, since I joined the Department of Cybernetics in 2017, the Department has always provided me with exceptional support. I would like to thank the Dept. Heads (originally Jan Kybic, later Tomáš Svoboda), Kristina Lukešová for support in HR and other matters, and Hana Pokorná in procurement. Special thanks go to Tomáš Svoboda for an absolutely unique combination of support, independence, and mentorship that he has granted me and my growing research group, and Petra Ivaničová who has been taking care of the budget of my research group and all administrative issues related to grant applications and management. Third, I want to thank to the reserarch groups where I studied and performed research in the past: the Artificial Intelligence Laboratory, Department of Informatics[1], University of Zurich (2006-2013) and Italian Institute of Technology (IIT, 2013-2016). The PhD study at the Zurich AI Lab under Rolf Pfeifer's supervision has provided a foundation for my career. Postdoctoral research at IIT and collaboration in particular with Alessandro Roncone and Ugo Pattacini constitutes another foundation for my current research. Fourth, I want to thank my external collaborators, in particular Kevin O'Regan, Jeffrey J. Lockman, Lisa Chinn, and Daniela Corbetta. Finally, I need to thank my wife Veronika, my daughters Anna and Tereza, and my parents.

The research reviewed in this thesis would not have been possible without the generous funding from:

- 2020-2024 – Czech Science Foundation (GA CR): "Whole-body awareness for safe and natural interaction: from brains to collaborative robots". Project no. 20-24186X.
- 2017-2019 – Czech Science Foundation (GA CR): "Robot self-calibration and safe physical human-robot interaction inspired by body representations in primate brains". Project no. 17-15697Y.
- 2018-2022 – The Ministry of Education, Youth and Sports of the Czech Republic, OP VVV funded project CZ.02.1.01/0.0/0.0/16_019/0000765 "Research Center for Informatics".
- 2014-2016 – Marie Curie Intra-European Fellowship (FP7-PEOPLE-2013-IEF 625727): "iCub Body Schema".

[1] The template used for this thesis is one of Department of Informatics.

# Abstract

Humans and animals excel in combining information from multiple sensory modalities, controlling their complex bodies, adapting to growth, failures, or using tools. These capabilities are also highly desirable in robots. They are displayed by machines to some extent. Yet, the artificial creatures are lagging behind. The key foundation is an internal representation of the body that the agent—human, animal, or robot—has developed. The mechanisms of operation of body models in the brain are largely unknown and even less is known about how they are constructed from experience after birth. In collaboration with developmental psychologists, we conducted targeted experiments to understand how infants acquire first "sensorimotor body knowledge". These experiments inform our work in which we construct embodied computational models on humanoid robots that address the mechanisms behind learning, adaptation, and operation of multimodal body representations. At the same time, we assess which of the features of the "body in the brain" should be transferred to robots to give rise to more adaptive and resilient, self-calibrating machines. We extend traditional robot kinematic calibration focusing on self-contained approaches where no external metrology is needed: self-contact and self-observation. Problem formulation allowing to combine several ways of closing the kinematic chain simultaneously is presented, along with a calibration toolbox and experimental validation on several robot platforms. Finally, next to models of the body itself, we study peripersonal space—the space immediately surrounding the body. Again, embodied computational models are developed and subsequently, the possibility of turning these biologically inspired representations into safe human-robot collaboration is studied.

# Contents

Chapter 1

# Introduction

The main theme in my research has been embodied cognition and its development, how it can be studied using robots, and how inspiration from biology can drive progress in robotic technology. The specific focus is on representations of the body and the space around it—peripersonal space. This habilitation thesis is constituted by a collection of thirteen articles (published or accepted for publication), preceded by a common introductory part that links these works into a coherent whole.

## 1.1 Synthetic methodology

The methodology employed in my work is the so-called *synthetic methodology*, or "understanding by building" [Pfeifer and Scheier, 2001, Chapter 1] [Pfeifer and Bongard, 2007, Chapter 3]. This approach can be applied to study phenomena from many disciplines like biology, psychology, or physics. As a matter of fact, many sciences that were traditionally mostly analytical are becoming more synthetic, employing computer simulations, for example. Here we focus on the understanding of intelligence or cognition in particular. The synthetic methodology, schematically illustrated in Fig. 1.1, unites the following three goals [Pfeifer and Bongard, 2007, Chapter 3]:

1. understanding natural forms of intelligence
2. abstracting general principles of intelligent behavior
3. building intelligent artifacts

A very influential paradigm in the study of cognition and intelligence was cognitivism (e.g., [Fodor, 1975,Pylyshyn, 1984]), whereby thinking was understood as a result of computation over symbols that represent the world. Such computation in the form of automatic manipulation of abstract symbols was at the heart of the so-called "Good Old-Fashioned Artificial Intelligence" (GOFAI) [Haugeland, 1985]. In fact, the computer was not only a tool for modeling but became a metaphor for the mind. Cognitivism and GOFAI were thus employing the full synthetic methodology in (1) applying the computer metaphor to model the mind; (2) abstracting general principles and developing algorithms for planning, reasoning, etc.; (3) developing applications like chess-playing programs, expert systems etc.

### 1.1.1 Embodied computational modeling of cognition

While sucessful in the abstract domains, GOFAI faced severe difficulties when controlling robots that had to interact with the physical world in real time. Rodney Brooks has openly attacked

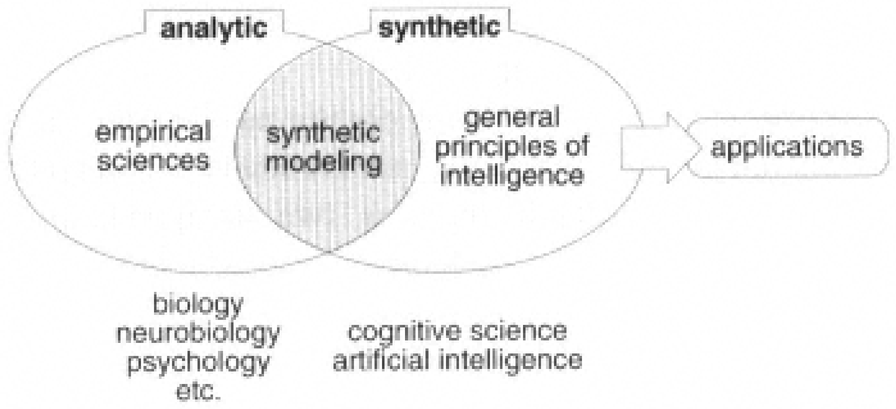


Figure 1.1: Synthetic methodology. Overview of approaches to the study of intelligence. On the left, we have the empirical sciences like biology, neurobiology, and psychology that mostly follow an analytic approach. In the center, we have the synthetic ones, namely cognitive science and AI, which can either model natural agents (this is called synthetic modeling, the shaded area) or alternatively can simply explore issues in the study of intelligence without necessarily being concerned about natural systems. From this latter activity, industrial applications can be developed. Figure and caption from [Pfeifer and Scheier, 2001, Chapter 1].

the GOFAI position in the seminal articles "Intelligence without representation" [Brooks, 1991b] and "Intelligence without reason" [Brooks, 1991a]. Through building robots that interact with the real world, such as insect robots [Brooks, 1989], he realized that "when we examine very simple level intelligence we find that explicit representations and models of the world simply get in the way. It turns out to be better to use the world as its own model." [Brooks, 1991b] The thesis that intelligent behavior emerges from the dynamic interplay of brain, body and environment has also been articulated by the notion of *embodiment* (e.g., [Pfeifer and Scheier, 2001,Pfeifer and Bongard, 2007]) that also started to challenge cognitivism as the dominant paradigm in cognitive science. Different variants of embodied cognition theories have been articulated like the dynamic systems approach to the development of cognition and action [Thelen and Smith, 1994], grounded cognition [Barsalou, 2008], sensorimotor contingency theory [O'Regan and Noe, 2001], or enaction [Varela et al., 1991].

Embodied cognition theories stressed the constitutive role of the body and action for even high-level abstract thinking. Given the importance of the body and closed sensorimotor loops, the analytical and divide-and-conquer approach of empirical sciences studying specific phemonena in isolation is more difficult to apply. Instead, a more holistic approach should be adopted. Applying the synthetic methodology to embodied cognition means not only modeling cognitive processes but requires building complete artifacts interacting with the environment. Grey Walter [Walter, 1953] was the pioneer of this approach already before the era of computers and cognitivism building electronic machines with a minimal "brain" that displayed phototactic-like behavior. This was picked up by Valentino Braitenberg who built a whole series of two-wheeled vehicles of increasing complexity, as summarized in "Vehicles – Experiments in Synthetic Psychology" [Braitenberg, 1986]. Already the most primitive ones, in which sensors are directly connected to motors (exciting or inhibiting them), displayed sophisticated behaviors. The obvious tools for this type of modeling became robots, giving rise to cognitive developmental robotics [Asada et al., 2009,Cangelosi and Schlesinger, 2015]. Whenever the emphasis is on the compatibility with available knowledge on the anatomy and physiology of the brain structures underlying the behaviors of interest, the label neurorobotics is also used (e.g., [van der Smagt et al., 2016]).

Let us first focus on the goal of understanding cognition and its development through synthethic modeling. The perspective of some of the leading researchers in this field is offered in "The mechanics of embodiment: a dialog on embodiment and computational modeling" [Pezzulo et al., 2011]. This approach is also articulated by Caligiore et al. [Caligiore et al., 2010, Caligiore et al., 2014] and called *computational embodied neuroscience*, with the following goals or constraints for the model:

- accurately reproduce the behaviors observed in specific psychological experiments
- reproduce the learning processes alongside the final behavior
- use architectures and algorithms constrained by neuroscientific evidence
- the model should control an embodied agent

Certain aspects of a natural system are studied, abstracted, and finally reproduced in an artificial system, which is then subject to investigations. The behavior as well as developmental and learning processes leading to the final behavior of the artificial system are compared with the original natural system, serving as a model of the phenomenon. Not only the overt behavior but also the inner workings responsible for it are compared. The key question is what can embodied cognition learn from synthetic modeling. The key benefits, inspired by Cangelosi and Parisi [Cangelosi and Parisi, 2002], are:

1. Explicit and detailed expression of theory. If the theory is "embedded" in a robot, it has to be explicit, detailed, and complete because otherwise it could not be tested in the robot.
2. Alternative validation. Any theory should ultimately be validated by comparing it to the biological or psychological phenomenon of interest. However, after implementing it in a robot, one can verify if it generates the expected behavior—for example, whether a machine built according to a model of walking actually walks. Often, this will not be the case and several iterations improving the theory will follow.
3. New and detailed predictions. The behavior of the robot may not match with the original expectations or may display new unexpected characteristics that can be compared with existing empirical data or lead to the design of new experiments.
4. Virtual experimental laboratories. With embodied computational models, one typically has access to all variables, including internal variables (raw and preprocessed sensory values, internal states, motor commands, etc.). In addition, it is often relatively easy to manipulate these variables and thus simulate experiments that would otherwise be impossible or ethically unacceptable to perform (e.g., simulating sensory or motor impairments).

My account of "Robots as powerful allies for the study of embodied cognition from the bottom up" is provided in [Hoffmann and Pfeifer, 2018] and included in Appendix A.

### 1.1.2 Embodied computational modeling of body representations

Synthetic modeling can be also productively applied to the study of body and peripersonal space representations. Since the focus is on models of the body, the need for embodied models is even stronger. Humanoid robots possess morphologies—physical characteristics as well as sensory and motor apparatus—that are in some respects akin to human bodies and can thus be used to expand the domain of computational modeling by anchoring it to the physical environment and a physical body and allowing for instantiation of complete sensorimotor loops. The iCub humanoid robot [Metta et al., 2010] stands out in this respect: it has anthropomorphic proportions modeled after a 4-year old child and a corresponding set of sensory modalities including binocular vision, hearing, vestibular (inertial) sensing, proprioception, and the recent important addition: touch [Bartolozzi et al., 2016], making it the ideal platform to model the multimodal contributions to body representations. Various surveys on body schema in robotics [Hoffmann et al., 2010], exploration, body representations and internal simulation [Schillaci et al., 2016], "synthetic psychology of the self" [Prescott and Camilleri, 2019] and robot models to study the (en)active self [Lanillos et al., 2017,Nguyen et al., 2021] have been published.

Remarkable demonstrations of this approach are the models of foetus development from the laboratory of Yasuo Kuniyoshi at the University of Tokyo (e.g., [Yamada et al., 2016]). They also illustate the potential of these models as virtual experimental laboratories, showing for example the effect of non-uniform versus uniform tactile sensor distribution [Mori and Kuniyoshi, 2010] or uterine versus extrauterine environment [Yamada et al., 2016].

Finally, it is important to note that not only is embodiment constitutive for the nature and operation of body models in the brain (and hence humanoid robots should be used to study body representations), but also, paraphrasing Brooks [Brooks, 1991b] the body may sometimes be its own best model. In other words, the physical body may be used directly, without having to have it represented in the brain (see "Embodying the mind and representing the body" [Alsmith and De Vignemont, 2012] or "body-in-the-brain versus body-in-the-world" [Ataria et al., 2021, p. XV-XVI] for a deeper discussion). Thus, with computational modeling only, one can hardly adopt other than the representationalist approach to body maps [De Vignemont, 2018, Chapter 5]. Instead, with robots, the sensorimotor approach to body know-how (e.g., [Jacquey et al., 2020]) can also be modeled.

### 1.1.3 Abstracting general principles of learning body models and applications

Apart from synthetic modeling of the mechanims of the "body in the brain", robots autonomously learning their body maps serve also abstracting general principles and even applications (see Fig. 1.1 again for the different aspects of the synthetic methodology). General characteristics of body models in animals, humans, and robots are derived in [Hoffmann, 2021b], included in Appendix B. The applications of robots learning their body models are in automatic self-contained robot self-calibration and they are reviewed in [Hoffmann, 2021a] and included in Appendix C.

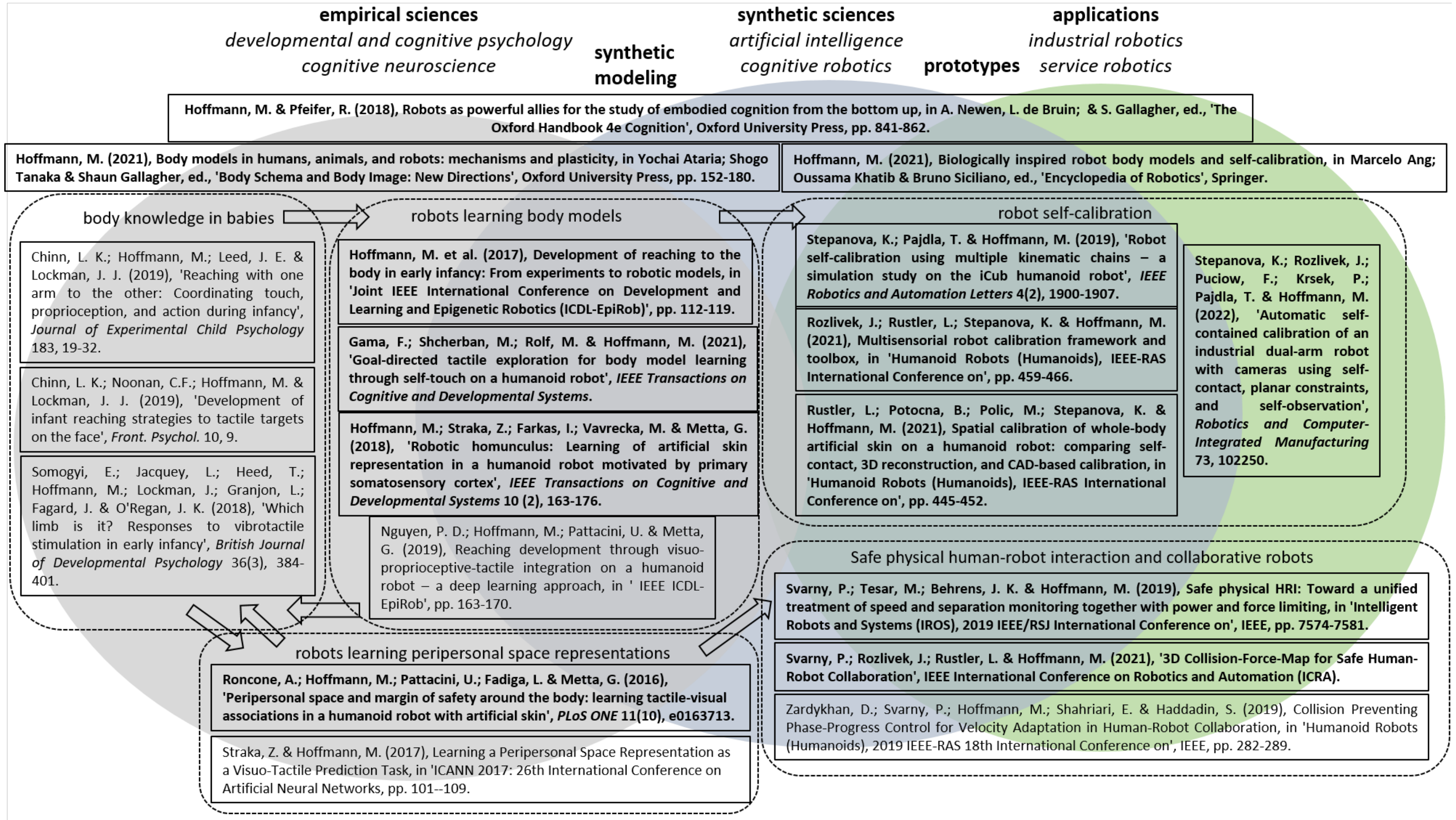
## 1.2 Body models in humans, animals, and robots

A theoretical account contrasting the body representations in the octopus, humans, and robots is presented in [Hoffmann, 2021b], included in Appendix B. On one hand, the example of the models of the iCub robot (like forward and inverse kinematics, inverse dynamics, etc.) and how it reaches for a tactile stimulus in [Roncone et al., 2014] serves as an intuition pump to analyze body models in animals and humans. On the other hand, what is known about how animals use and represent their bodies, including the imporant aspect that behavior generation is in the brain-body-environment coupling and hence not all features need to be modeled, is used to inspire the design of future generations of machines that would inherit some of the capacities demonstrated by animals like resilience.

## 1.3 Thesis overview

This thesis is a collection of published works, complemented by relatively brief texts that explain the relationships between the individual articles. The story of the thesis is largely told in three book chapters and the reader is advised to consult them. In [Hoffmann and Pfeifer, 2018], Appendix A, the methodology of using humanoid robots to study embodied cognition is explained and illustrated on examples. In [Hoffmann, 2021b], Appendix B, body models in humans, animals, and robots are contrasted. In [Hoffmann, 2021a], Appendix C, an overview of biologically inspired robot body models and self-calibration is provided for a robotics encyclopedia. All three articles were prepared for a wider, often interdisciplinary, audience, and they are thus accessible even for non-experts in a specific field.

The rest of the thesis brings together articles with a more technical content and groups them as follows. In Chapter 2, an overview of related research on the development of body representations in children is provided, along with my own contributions through collaborations with developmental psychologists. Chapter 3 describes biologically motivated case studies on robots learning body models. Chapter 4 focuses on robots learning about the space surrounding their bodies, so called peripersonal space. Chapter 5 deals with the topic of automatic self-contained robot calibration. Chapter 6 brings together works on safety of physical human-robot interaction. The thesis Appendix includes thirteen peer-reviewed published works pertinent to the thesis topic. With one exception [Roncone et al., 2016], I am either the first or the senior author on all these publications—in the latter case, they originate from my research group.



**empirical sciences**
*developmental and cognitive psychology*
*cognitive neuroscience*

**synthetic modeling**

**synthetic sciences**
*artificial intelligence*
*cognitive robotics*

**prototypes**

**applications**
*industrial robotics*
*service robotics*

**Hoffmann, M. & Pfeifer, R. (2018), Robots as powerful allies for the study of embodied cognition from the bottom up, in A. Newen, L. de Bruin; & S. Gallagher, ed., 'The Oxford Handbook 4e Cognition', Oxford University Press, pp. 841-862.**

**Hoffmann, M. (2021), Body models in humans, animals, and robots: mechanisms and plasticity, in Yochai Ataria; Shogo Tanaka & Shaun Gallagher, ed., 'Body Schema and Body Image: New Directions', Oxford University Press, pp. 152-180.**

**Hoffmann, M. (2021), Biologically inspired robot body models and self-calibration, in Marcelo Ang; Oussama Khatib & Bruno Siciliano, ed., 'Encyclopedia of Robotics', Springer.**

body knowledge in babies

Chinn, L. K.; Hoffmann, M.; Leed, J. E. & Lockman, J. J. (2019), 'Reaching with one arm to the other: Coordinating touch, proprioception, and action during infancy', *Journal of Experimental Child Psychology* 183, 19-32.

Chinn, L. K.; Noonan, C.F.; Hoffmann, M. & Lockman, J. J. (2019), 'Development of infant reaching strategies to tactile targets on the face', *Front. Psychol.* 10, 9.

Somogyi, E.; Jacquey, L.; Heed, T.; Hoffmann, M.; Lockman, J.; Granjon, L.; Fagard, J. & O'Regan, J. K. (2018), 'Which limb is it? Responses to vibrotactile stimulation in early infancy', *British Journal of Developmental Psychology* 36(3), 384-401.

robots learning body models

**Hoffmann, M. et al. (2017), Development of reaching to the body in early infancy: From experiments to robotic models, in 'Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)', pp. 112-119.**

**Gama, F.; Shcherban, M.; Rolf, M. & Hoffmann, M. (2021), 'Goal-directed tactile exploration for body model learning through self-touch on a humanoid robot', *IEEE Transactions on Cognitive and Developmental Systems*.**

**Hoffmann, M.; Straka, Z.; Farkas, I.; Vavrecka, M. & Metta, G. (2018), 'Robotic homunculus: Learning of artificial skin representation in a humanoid robot motivated by primary somatosensory cortex', *IEEE Transactions on Cognitive and Developmental Systems* 10 (2), 163-176.**

Nguyen, P. D.; Hoffmann, M.; Pattacini, U. & Metta, G. (2019), Reaching development through visuo-proprioceptive-tactile integration on a humanoid robot – a deep learning approach, in ' IEEE ICDL-EpiRob', pp. 163-170.

robot self-calibration

**Stepanova, K.; Pajdla, T. & Hoffmann, M. (2019), 'Robot self-calibration using multiple kinematic chains – a simulation study on the iCub humanoid robot', *IEEE Robotics and Automation Letters* 4(2), 1900-1907.**

**Rozlivek, J.; Rustler, L.; Stepanova, K. & Hoffmann, M. (2021), Multisensorial robot calibration framework and toolbox, in 'Humanoid Robots (Humanoids), IEEE-RAS International Conference on', pp. 459-466.**

**Rustler, L.; Potocna, B.; Polic, M.; Stepanova, K. & Hoffmann, M. (2021), Spatial calibration of whole-body artificial skin on a humanoid robot: comparing self-contact, 3D reconstruction, and CAD-based calibration, in 'Humanoid Robots (Humanoids), IEEE-RAS International Conference on', pp. 445-452.**

**Stepanova, K.; Rozlivek, J.; Puciow, F.; Krsek, P.; Pajdla, T. & Hoffmann, M. (2022), 'Automatic self-contained calibration of an industrial dual-arm robot with cameras using self-contact, planar constraints, and self-observation', *Robotics and Computer-Integrated Manufacturing* 73, 102250.**

robots learning peripersonal space representations

**Roncone, A.; Hoffmann, M.; Pattacini, U.; Fadiga, L. & Metta, G. (2016), 'Peripersonal space and margin of safety around the body: learning tactile-visual associations in a humanoid robot with artificial skin', *PLoS ONE* 11(10), e0163713.**

Straka, Z. & Hoffmann, M. (2017), Learning a Peripersonal Space Representation as a Visuo-Tactile Prediction Task, in 'ICANN 2017: 26th International Conference on Artificial Neural Networks, pp. 101--109.

Safe physical human-robot interaction and collaborative robots

**Svarny, P.; Tesar, M.; Behrens, J. K. & Hoffmann, M. (2019), Safe physical HRI: Toward a unified treatment of speed and separation monitoring together with power and force limiting, in 'Intelligent Robots and Systems (IROS), 2019 IEEE/RSJ International Conference on', IEEE, pp. 7574-7581.**

**Svarny, P.; Rozlivek, J.; Rustler, L. & Hoffmann, M. (2021), '3D Collision-Force-Map for Safe Human-Robot Collaboration', IEEE International Conference on Robotics and Automation (ICRA).**

Zardykhan, D.; Svarny, P.; Hoffmann, M.; Shahriari, E. & Haddadin, S. (2019), Collision Preventing Phase-Progress Control for Velocity Adaptation in Human-Robot Collaboration, in 'Humanoid Robots (Humanoids), 2019 IEEE-RAS 18th International Conference on', IEEE, pp. 282-289.

Figure 1.2: Habilitation Overview. Schematic visualization of the articles composing the thesis, grouped according to common themes, and overlayed on the synthetic methodology (analogous to Fig. 1.1). The articles in bold letters are included in the thesis appendices.

# Chapter 2

# Body knowledge in babies

A wealth of observations has been accumulated about development of the foetus and the infant in the months before and after birth. There is certainly a great deal to learn for the infant in this period, but the assumption in my work is that the, possibly primary, target of what the infant is learning about in this period is its own body. Understanding what its body parts are, what it can do with them, and what the consequences would be, are essential prerequisites for almost everything that follows. This is the sensorimotor or ecological self [Neisser, 1988]. The second assumption relates to the way in which the infant learns. In my work, self-touch or haptic exploration of one's own body is considered a key activity that may bootstrap learning about the body as an object in space [Hoffmann, 2017]. We assume that motor-proprioceptive-tactile contingencies provide sufficient material for the infant to construct "functional body knowledge", or body know-how, allowing her for example to reach for specific body parts when presented with a tactile stimulus. This sensorimotor account is presented in [Hoffmann et al., 2017] and included in Appendix D. The role of vision is increasingly important during postnatal development as the acuity and gaze control improves. Eventually, it is likely that the representations of the body and peripersonal space become predominantly vision-based. However, to what extent, when, and how this occurs still needs to be established.

## 2.1 Spontaneous self-touch

[1] Fetuses initially perform local movements directed to areas of the body most sensitive to touch: the face, but also soles of feet [Piontelli, 2015, p. 113-114]. Later, from 26 to 28 weeks of gestational age, they also use the back of the hands and touch other body areas like thighs, legs, and knees [Piontelli, 2015, p. 29-30]. In addition, from 19 weeks, fetuses anticipate hand-to-mouth movements [Myowa-Yamakoshi and Takeshita, 2006] (the mouth opens prior to contact) and from 22 weeks, the movements seem to show the recognizable form of intentional actions, with kinematic patterns that depend on the goal of the action (toward the mouth vs. toward the eyes) [Zoia et al., 2007].

Hand-mouth coordination continues to develop after birth [Rochat, 1993]. Specifically related to body exploration, [Rochat, 1998] writes: "By 2-3 months, infants engage in exploration of their own body as it moves and acts in the environment. They babble and touch their own body, attracted and actively involved in investigating the rich intermodal redundancies, temporal contingencies, and spatial congruence of self-perception." [DiMercurio et al., 2018] followed spontaneous behavior of infants from 3 to 9 weeks; [Thomas et al., 2015] from birth to 6 months of age. Their main findings regarding self-touch were:

[1] This section is adapted from Section II in [Gama et al., 2021].

- rostro-caudal progression: head and trunk contacts are more frequent in the beginning, followed by more caudal body locations including hips, then legs, and eventually the feet [Thomas et al., 2015]

- contacts are typically made with the ipsilateral hand

- *complex touches*, as infants moved their hand while remaining in contact with their body, were frequently observed by [DiMercurio et al., 2018]

In summary, infants acquire ample experience of touching their body, which allows for the learning of the first tactile-proprioceptive-motor models of the body. The ability to learn from this experience goes hand in hand with dynamic neural development in this period [Tau and Peterson, 2010]; see [Hoffmann, 2017] for a review focusing specifically on self-touch. Yet, the behavioral organization of such early tactile exploration is not understood. Are the touches on the body spontaneous or systematic? If there is a particular structure—which seems to be the case [DiMercurio et al., 2018,Thomas et al., 2015]—what drives this developmental progression? [Piaget, 1952] theorized that in newborns, action and perception as well as the "spaces" of individual sensory modalities are separated (cf. [Van der Meer et al., 1995] for evidence that visual and motor modalities are connected early after birth). Until the connections are established, infants explore their environment (and their body) randomly. [Piaget, 1952] also proposed a pivotal role of repeated movements—*primary circular reactions* directed to learn properties of the body and *secondary circular reactions* driven by the interest on the effects they produce in the environment. However, to discriminate spontaneous contacts from systematic (intrinsically motivated) exploration remains a challenge.

## 2.2 Localizing touch on the body

[2] A counterpart to recordings of spontaneous infant behavior is provided by testing how they can reach to targets on their body. Lockman and colleagues performed a series of studies [Chinn et al., 2019, Chinn et al., 2021, Hoffmann et al., 2017, Leed et al., 2019, Somogyi et al., 2018] in which vibrotactile targets ("buzzers") were attached to infants' body parts and their ability and their way of reaching for the targets were analyzed. Targets above the mouth and on the chin were successfully contacted already from 2 months of age [Chinn et al., 2021], followed by trunk area, legs, hands, other areas on the face (forehead, ears), and elbows (around 9 months) (whole body – pilot study [Hoffmann et al., 2017]; upper body [Leed et al., 2019]). For targets on hands and arms, the arm with the buzzer and the contralateral arm reaching for the target often moved simultaneously—the arm with the target actively facilitating the removal [Chinn et al., 2019]. Regarding looking at the target for locations where this was possible, the infants looked first in 86 of 189 trials (45.5%), reached first in 27 of 189 trials (14.29%), and looked and reached simultaneously in 76 of 189 trials (40.21%). This last strategy significantly increased with age (7 to 21 months).

Another window into the development of tactile localization is provided by experiments that exploit crossed hands or crossed feet postures, often employed to study conflicts between encoding touch in an anatomical versus external frame of reference. In short, when one's left hand is located in the right peripersonal space, there is a conflict between where the hand is normally and currently, which impairs tactile localization (one manifestation is slower response in the temporal order judgement task – [Heed and Azañón, 2014] for a review). [Ali et al., 2015] tested two groups of infants—four-month and six-month-olds—by applying tactile stimulation to their feet in a crossed posture. Six-month-olds, like adults, showed a tactile localisation deficit, indicating

[2] The first paragraph is adapted from Section II in [Gama et al., 2021].

external spatial coding of touch; in striking contrast, four-month-olds outperformed the older infants showing no crossed-feet deficit, indicating that coding of touch in external space, called also tactile remapping [Heed et al., 2015], develops at around 5 months after birth, possibly in line with the growing importance of vision. In another study, [Bremner et al., 2008] tested 6.5-month- and 10-month-olds in a crossed and uncrossed hand posture and investigated their manual and looking responses.

## 2.3 My contributions

The studies briefly reviewed above provide important constraints for the embodied computational modeling work (Chapter 3). Some of the authors of the works cited above have become my collaborators (Prof. Jeffrey J. Lockman, Tulane University, USA; Prof. Daniela Corbetta, University of Tennessee, Knoxville, USA; Dr. Kevin O'Regan, University Paris Descartes & CNRS, Paris, France; Prof. Tobias Heed, Paris Lodron University Salzburg, Austria). I have co-authored the following articles:

- Chinn, L. K., Hoffmann, M., Leed, J. E., & Lockman, J. J. (2019). Reaching with one arm to the other: coordinating touch, proprioception, and action during infancy. *Journal of experimental child psychology*, 183, 19-32. [M.H. author contribution 25%]

- Chinn, L. K., Noonan, C. F., Hoffmann, M., & Lockman, J. J. (2019). Development of infant reaching strategies to tactile targets on the face. *Frontiers in psychology*, 10, 9. [M.H. author contribution 20%]

- Somogyi, E., Jacquey, L., Heed, T., Hoffmann, M., Lockman, J. J., Granjon, L., Fagard, J., & O'Regan, J. K. (2018). Which limb is it? Responses to vibrotactile stimulation in early infancy. *British Journal of Developmental Psychology*, 36(3), 384-401. [M.H. author contribution 10%]

These articles are not included in this habilitation which is submitted for the Technical Cybernetics program.

In addition, part II.C of [Hoffmann et al., 2017], Appendix D, and part II of [Gama et al., 2021], Appendix E contain also sections reporting results of experiments with infants specifically targeting the needs of the modeling endeavour.

## 2.4 Summary

The observations from the studies on babies provide inputs and constraints for the embodied computational modeling. In line with the synthetic methodology (Section 1.1), inspiration from biology may contribute to the discovery of general principles that may eventually give rise to useful technology like self-calibrating robots. However, the primary goal is to help uncover the mechanims of the development of body know-how in infants. Collecting unambiguous empirical evidence is complicated by the age of the subjects—infants in the first year are preverbal and one thus cannot easily instruct them and their cooperation during experiments is limited. Similarly, manipulations (e.g., reaching for a tactile stimulus only with a certain limb) are hard to orchestrate and additional setups like motion capture or brain imaging are also almost impossible to arrange. Therefore, robot models that allow arbitrary manipulations and where all internal variables can be accessed provide an indispensable tool.

# Chapter 3

# Robots learning body models

The results of empirical studies from Chapter 2 flow into embodied computational models on humanoid robots that address the mechanisms of development of reaching and somatosensory perception in early infancy. Humanoid robots with pressure-sensitive electronic skins covering large areas of their bodies provide the right platform for this type of work. There are four articles in this research strand [Hoffmann et al., 2017, Gama et al., 2021, Hoffmann et al., 2018, Nguyen et al., 2019], three of which are included in the thesis.

The article [Hoffmann et al., 2017], included in Appendix D, is a conceptual one, in collaboration with developmental and cognitive psychologists and provides a natural transition from the empirical studies to embodied computational modeling of the development of reaching to the body.

The second article [Gama et al., 2021], included in Appendix E, focuses on active body exploration and compares different algorithms from the family of intrinsically motivated learning (or artificial curiosity) [Schmidhuber, 1991, Oudeyer and Kaplan, 2007, Baranes and Oudeyer, 2013, Baldassarre and Mirolli, 2013]. This work demonstrates that efficient exploration of the skin space and learning of inverse models is possible. First grounding of the modeling work in experimental data is attempted (study of motor redundancy in infant reaching and link to "complex touches" from [DiMercurio et al., 2018]).

The third article [Hoffmann et al., 2018], included in Appendix F, studies the development of a biologically motivated representation of the robot skin surface: the robot "tactile homunculus". The robot is exposed to tactile stimulations on its whole body and the corresponding tactile activations are recorded. These are then fed into a self-organizing (or Kohonen) map algorithm and the representations that emerge are studied. Modifications of the standard algorithm that provide the right constraints to channel learning toward the layout of the neural map observed in primate brains are developed.

The fourth article [Nguyen et al., 2019] studies the development of reaching to objects external to the body. These are perceived visually, but we concentrate on the role of haptic feedback in learning the behavior on the robot. If the object is at first randomly contacted, proprioception provides an alternative to vision to guide subsequent reaching movements. Such "somatosensory coding of space" connects reaching to the body with reaching to external objects.

# Chapter 4

# Robots learning peripersonal space representations

Next to "body space" or "personal space", the space immediately surrounding the body is called peripersonal space. There are two related but different meanings associated with peripersonal space (PPS): (i) space immediately surrounding the body and (ii) space that we can act upon / within our reach. Their representations in the brain are realized by fronto-parietal networks, with an important role attributed to bimodal neurons with visuo-tactile receptive fields (RFs) [Cléry et al., 2015]. The first notion, space surrounding the body, can be pictured as a "bubble" around individual body parts and following those body parts in space. It is realized by bimodal neurons with tactile RFs on the skin and visual RFs around. The visual responses appear to be tuned to dynamically approaching objects and their activation is thus anticipatory, predictive of touch in the corresponding skin area. There are also behavioral responses associated with the stimulation of some neurons of this network (squinting, ducking, and withdrawing from the direction of the potential threat). Therefore, this circuit is thought to be responsible for self-defense and maintaining a safety margin around the body [Graziano and Cooke, 2006]. Little is known about the development of peripersonal space representations, with some evidence suggesting that even newborns can make sense of multisensory (audio-visual in this case) cue combinations specifying motion with respect to themselves [Orioli et al., 2018].

Taking advantage of humanoid robots with whole-body sensitive skin, we designed a method where the robot learns such a safety margin from experience. The robot records trajectories of approaching objects and if they eventually contact its body as perceived by the artificial skin, it learns the likelihood of such future contact. The objects' positions perceived by the robot by stereo vision are remapped into the reference frames of every tactile sensor (taxel). Every taxel then continuously updates a representation of the probability of objects at a certain distance [Roncone et al., 2015] or with a certain time to contact [Roncone et al., 2016] colliding with it. These "threatened" taxels can be aggregated and a most likely future collision site on the robot body computed. This location on the robot body can then be connected to a controller that avoids the collision by moving the exposed body part away (video: `https://youtu.be/3IaXxNwC_7E`). Alternatively, using the opposite sign for the movement direction, "whole-body reaching/catching" can be easily realized. The work [Roncone et al., 2016] is included in Appendix G.

We developed an alternative learning algorithm in [Straka and Hoffmann, 2017], a neural network composed of a Restricted Boltzmann Machine and a feedforward neural network. The former learns in an unsupervised manner to represent position and velocity features of the stimulus. The latter is trained in a supervised way to predict the position of touch (contact). Compared to [Roncone et al., 2016], this model was trained in a simulated environment but it importantly take into account also the the uncertainty of all variables.

Apart from modeling the operation, development, and adaption of peripersonal space representations in the brain, “perirobot space” representations are key for safe human-robot interaction. Appropriate collision avoidance strategies cannot be selected without a representation of the space surrounding the robot. This space can be represented in various forms with regards to the needs of the application or robot platform. Multiple questions come forth in this context, namely the way the space is structured, how it will dynamically adapt and what is its geometry. A key component is the form of the representation of the robot and human body parts, or, in general, the representation of obstacles. Drawing on the results of the computer graphics community (see [Jiménez et al., 2001] for a survey), this often takes the form of some collision primitives. These can be simple shapes like spheres [Flacco et al., 2012] or more complex meshes [Polverini et al., 2017] and can differ for the robot and the human. [Zanchettin et al., 2015] represent robot links as segments and humans as a set of capsules. These shapes can also have a temporal aspect and represent so-called swept volumes, i.e. zones where the human or robot moved. For safety to be guaranteed, the whole body of both agents should be represented and considering only the robot end-effector or human hands or head does not suffice. Some representations change the volume dynamically based on the robot or human velocity [Polverini et al., 2017, Zanchettin et al., 2015, Lacevic and Rocco, 2010, Magnanimo et al., 2016]. The technologies applied to perception can influence the structure of the perirobot space too: Euclidean space can be replaced by a depth space approach to account for the occlusions and specific geometries of the field of view of a RGB-D sensor [Flacco et al., 2012, Flacco et al., 2015]. Examples are depicted in Fig. 4.1. The best representation is yet to be found. These considerations tie directly into the topic of safe physical human-robot interaction discussed in Chapter 6 and our publication [Svarny et al., 2019].

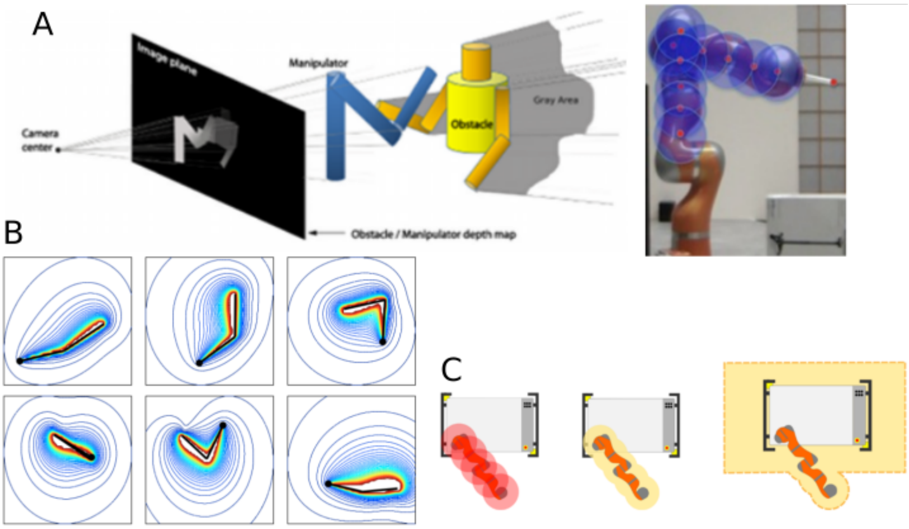


Figure 4.1: Perirobot space representations. (A) Depth space approach [Flacco et al., 2012]. (B) Kinetostatic danger field [Lacevic and Rocco, 2010]. (C) Dynamic safety fields [Magnanimo et al., 2016].

# Chapter 5

# Robot self-calibration

As briefly reviewed in Chapter 2, newborns have not only limited conceptual and spatial knowledge about their bodies, but also limited "body know-how"—means to use their bodies for purposeful action or to localize and act on stimuli on their body. They learn the necessary body models in the first years after birth, while continuously incorporating physical body growth as well as maturation of the sensory apparatus (vision in particular). Next to the developmental time scale, human body representations were found to be adaptive (called *plastic* in neuroscience) on much shorter time scales, as demonstrated by the ability to use tools and their incorporation into the body schema [Maravita and Iriki, 2004], for example. Both capacities—learning and quickly adapting body models—would be highly desired in robots.

Nowadays, humanoid but also other robots come with a rich set of powerful yet inexpensive sensors like cameras, RGB-D cameras, inertial, tactile or force sensors. This opens up the possibility for calibration approaches that are more self-contained, can be performed autonomously and repeatedly by the robot, and that simultaneously estimate the position of the sensors with respect to the robot. The key to self-calibration is *redundancy*. The kinematic chain can be closed exploiting physical contact (aka *closed-loop calibration* approaches) or by observing the robot pose using visual sensors (*open-loop calibration* approaches) (see [Hollerbach et al., 2016]). Fig. 5.1 provides an overview. Next to traditional methods exploiting contact with the environment (e.g. robot touching a planar surface – Fig. 5.1B) or external metrology systems (e.g. laser trackers – Fig. 5.1D), we studied self-contact (Fig. 5.1A) and self-observation (Fig. 5.1C) as methods that are suited for automatic self-contained calibration.

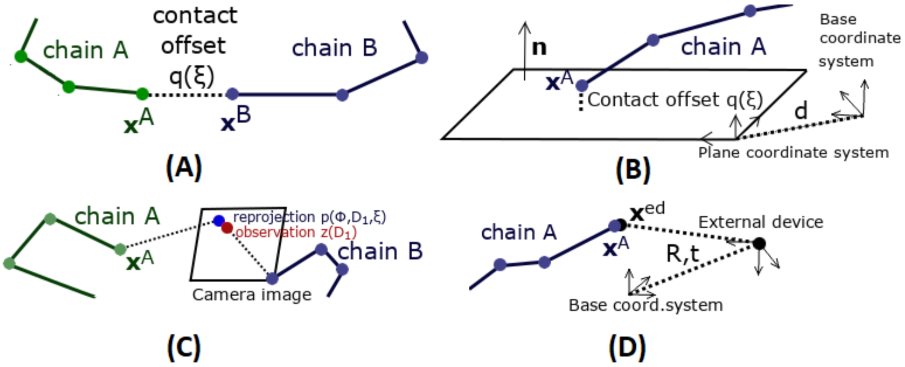


Figure 5.1: Schematics of calibration using self-contact (A), contact with a plane (B), self-observation (C), and external device (D). Figure and caption from [Rozlivek et al., 2021].

An overview of the state of the art and future research directions regarding biologically inspired robot body models and self-calibration is provided in [Hoffmann, 2021a], included in Appendix C. In addition, four case studies are also included in this thesis.

In [Stepanova et al., 2019], Appendix H, we systematically studied on the simulated iCub humanoid robot how self-observation, self-contact, and their combination can be used for self-calibration. We found that employing multiple kinematic chains (self-observation and self-touch) is superior in terms of optimization results as well as observability.

In [Rozlivek et al., 2021], Appendix I, we provided a unified formulation that makes it possible to combine traditional approaches with self-contained calibration available to humanoid robots in a single framework and single cost function. Second, we presented an open source toolbox for Matlab (`https://github.com/ctu-vras/multirobot-calibration`) that provides this functionality, along with additional tools for preprocessing (e.g., dataset visualization) and evaluation (e.g., observability/identifiability).

In [Rustler et al., 2021], Appendix J, we used self-contact as one of the methods to calibrate the positions of 970 pressure sensors on the body of a humanoid robot (the opposite approach to [Roncone et al., 2014] where the skin was used to improve kinematic calibration). We experimentally compared the accuracy and effort associated with the following skin spatial calibration approaches and their combinations: (i) combining CAD models and skin layout in 2D, (ii) 3D reconstruction from images, (iii) using robot kinematics to calibrate skin by self-contact.

Finally, in [Stepanova et al., 2022], Appendix K, all four calibration methods schematically illustrated in Fig. 5.1—self-contact, contact with a plane, self-observation, and external device—were experimentally compared on a dual-arm industrial manipulator. The main findings were: (1) when applying the complementary calibration approaches in isolation, the self-contact approach yields the best and most stable results; (2) all combinations of more than one approach were always superior to using any single approach in terms of calibration errors and the observability of the estimated parameters. Combining more approaches delivers robot parameters that better generalize to the workspace parts not used for the calibration.

Chapter 6

# Safe physical human-robot interaction and collaborative robots

Robots are leaving safety fences and start to share workspaces or even living spaces with humans. As they leave controlled environments and enter domains that are far less structured, they need to dynamically adapt to unpredictable interactions and guarantee safety at every moment. There has been rapid development in this regard in the last decade, with revisions of existing and introduction of new safety standards ( [ISO 10218, 2011, ISO 13855, 2010, ISO/TS 15066, 2016]; see e.g. [Haddadin and Croft, 2016] for a survey)) and a rapidly growing market of collaborative robots. According to [ISO/TS 15066, 2016], there are two ways of satisfying the safety requirements when a human physically collaborates with a robot: (i) *Power and Force Limiting (PFL)* and (ii) *Speed and Separation Monitoring (SSM)*.

For PFL, physical contacts with a moving robot are allowed but the forces / pressures / energy absorbed during a collision need to be within human body part specific limits. This translates onto lightweight structure, soft padding and no pinch points on the robot side, in combination with collision detection and response relying on motor load measurements, force/torque or joint torque sensing. This is addressed by interaction control methods for this post-impact phase (e.g., [De Luca et al., 2006,Haddadin et al., 2008]; [Haddadin et al., 2017] for a survey). The performance of robots complying with this safety requirement in terms of payload, speed, and repeatability is limited.

Safe collaborative operation according to speed and separation monitoring prohibits contacts with a moving robot and thus focuses on the pre-impact phase: a protective separation distance, $S_p$, between the operator and robot needs to be maintained at all times. When the distance decreases below $S_p$, the robot is commanded to halt. In industry, $S_p$ is typically safeguarded using light curtains (essentially electronic versions of physical fences) or safety-rated scanners that monitor 2D or 3D zones. However, the flexibility of such setups is limited—the information is reduced to detecting whether an object of a certain minimum volume has entered a predefined zone. The higher the robot kinetic energy, the bigger is its footprint on the factory floor.

This topic ties with the rest of the thesis as follows. First, not only robot performance but also safety is dependent on its accuracy and calibration. Thus, machines learning and adapting their body models (Chapter 3 or self-calibrating robots in Chapter 5) are also likely to be safer for their environments. Second, the use of artificial sensitive skins researched in my group can contribute an additional protective layer for collision detection and isolation during physical HRI (the PFL regime in particular). Third, adaptive multimodal representations of the robot peripersonal space (Chapter 4) could give rise to robots with a human-like margin of safety around their bodies and to new solutions to robot safety in the SSM regime.

In [Svarny et al., 2019], Appendix L, we deployed the two collaborative regimes (PFL and

SSM) in a single application and studied the performance in a mock collaborative task under the individual regimes, including transitions between them. Additionally, we compared the performance under “safety zone monitoring” with keypoint pair-wise separation distance assessment relying on an RGB-D sensor and skeleton extraction algorithm to track human body parts in the workspace. Best performance has been achieved in the following setting: robot operates at full speed until a distance threshold between any robot and human body part is crossed; then, reduced robot speed per power and force limiting is triggered. Robot is halted only when the operator’s head crosses a predefined distance from selected robot parts.

In [Svarny et al., 2021], Appendix M, we measured the forces exerted by two collaborative manipulators moving downward against an impact measuring device. First, we empirically showed that the impact forces can vary by more than 100 percent within the robot workspace. The forces are negatively correlated with the distance from the robot base and the height in the workspace. Second, we presented a data-driven model, 3D Collision-Force-Map, predicting impact forces from distance, height, and velocity and demonstrate that it can be trained on a limited number of data points. Third, we analyzed the force evolution upon impact and found that clamping never occurs for one of the robots (UR10e). We showed that formulas relating robot mass, velocity, and impact forces from ISO/TS 15066 [ISO/TS 15066, 2016] are insufficient—leading both to significant underestimation and overestimation and thus to unnecessarily long cycle times or even dangerous applications. We proposed an empirical method that can be deployed to quickly determine the optimal speed and position where a task can be safely performed with maximum efficiency.

Finally, in collaboration with the group of Sami Haddadin at Technical University of Munich, we investigated how the velocity of a collaborative manipulator can be modulated by monitoring the interaction with a human. The details are in [Zardykhan et al., 2019] (not included in Appendix).

# Bibliography

[Ali et al., 2015] Ali, J. B., Spence, C., and Bremner, A. J. (2015). Human infants' ability to perceive touch in external space develops postnatally. *Current Biology*, 25(20):R978–R979.

[Alsmith and De Vignemont, 2012] Alsmith, A. J. T. and De Vignemont, F. (2012). Embodying the mind and representing the body. *Review of Philosophy and Psychology*, 3(1):1–13.

[Asada et al., 2009] Asada, M., Hosoda, K., Kuniyoshi, Y., Ishiguro, H., Inui, T., Yoshikawa, Y., Ogino, M., and Yoshida, C. (2009). Cognitive developmental robotics: a survey. *IEEE Transactions on Autonomous Mental Development*, 1(1):12–34.

[Ataria et al., 2021] Ataria, Y., Tanaka, S., and Gallagher, S. (2021). *Body Schema and Body Image: New Directions*. Oxford University Press.

[Baldassarre and Mirolli, 2013] Baldassarre, G. and Mirolli, M. (2013). *Intrinsically motivated learning in natural and artificial systems*. Springer.

[Baranes and Oudeyer, 2013] Baranes, A. and Oudeyer, P.-Y. (2013). Active learning of inverse models with intrinsically motivated goal exploration in robots. *Robotics and Autonomous Systems*, 61(1):49–73.

[Barsalou, 2008] Barsalou, L. (2008). Grounded cognition. *Annual Review of Psychology*, 59:617–645.

[Bartolozzi et al., 2016] Bartolozzi, C., Natale, L., Nori, F., and Metta, G. (2016). Robots with a sense of touch. *Nature materials*, 15(9):921–925.

[Braitenberg, 1986] Braitenberg, V. (1986). *Vehicles - Experiments in Synthetic Psychology*. MIT Press, Cambridge, MA.

[Bremner et al., 2008] Bremner, A. J., Mareschal, D., Lloyd-Fox, S., and Spence, C. (2008). Spatial localization of touch in the first year of life: early influence of a visual spatial code and the development of remapping across changes in limb position. *Journal of Experimental Psychology: General*, 137(1):149.

[Brooks, 1989] Brooks, R. A. (1989). A robot that walks: Emergent behaviors from a carefully evolved network. *Neural Computation*, 1:153–162.

[Brooks, 1991a] Brooks, R. A. (1991a). Intelligence without reason. In *Proc. 12th Int. Joint Conf. on Artificial Intelligence*.

[Brooks, 1991b] Brooks, R. A. (1991b). Intelligence without representation. *Artificial Intelligence Journal*, 47:139–159.

[Caligiore et al., 2010] Caligiore, D., Borghi, A. M., Parisi, D., and Baldassarre, G. (2010). Tropicals: a computational embodied neuroscience model of compatibility effects. *Psychological Review*, 117(4):1188.

[Caligiore et al., 2014] Caligiore, D., Parisi, D., and Baldassarre, G. (2014). Integrating reinforcement learning, equilibrium points, and minimum variance to understand the development of reaching: a computational model. *Psychological review*, 121(3):389.

[Cangelosi and Parisi, 2002] Cangelosi, A. and Parisi, D. (2002). Computer simulation: A new scientific approach to the study of language evolution. In *Simulating the evolution of language*, pages 3–28. Springer.

[Cangelosi and Schlesinger, 2015] Cangelosi, A. and Schlesinger, M. (2015). *Developmental robotics*. MIT press.

[Chinn et al., 2019] Chinn, L. K., Hoffmann, M., Leed, J. E., and Lockman, J. J. (2019). Reaching with one arm to the other: Coordinating touch, proprioception, and action during infancy. *Journal of Experimental Child Psychology*, 183:19–32.

[Chinn et al., 2021] Chinn, L. K., Noonan, C. F., and Lockman, J. J. (2021). The human face becomes mapped as a sensorimotor reaching space during the first year. *Child Development*, 92(2):760–773.

[Cléry et al., 2015] Cléry, J., Guipponi, O., Wardak, C., and Hamed, S. B. (2015). Neuronal bases of peripersonal and extrapersonal spaces, their plasticity and their dynamics: knowns and unknowns. *Neuropsychologia*, 70:313–326.

[De Luca et al., 2006] De Luca, A., Albu-Schaffer, A., Haddadin, S., and Hirzinger, G. (2006). Collision detection and safe reaction with the dlr-iii lightweight manipulator arm. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 1623–1630. IEEE.

[De Vignemont, 2018] De Vignemont, F. (2018). *Mind the body: An exploration of bodily self-awareness*. Oxford University Press.

[DiMercurio et al., 2018] DiMercurio, A., Connell, J. P., Clark, M., and Corbetta, D. (2018). A naturalistic observation of spontaneous touches to the body and environment in the first 2 months of life. *Frontiers in psychology*, 9:2613.

[Flacco et al., 2015] Flacco, F., Kroeger, T., De Luca, A., and Khatib, O. (2015). A depth space approach for evaluating distance to objects. *Journal of Intelligent & Robotic Systems*, 80(1):7–22.

[Flacco et al., 2012] Flacco, F., Kröger, T., De Luca, A., and Khatib, O. (2012). A depth space approach to human-robot collision avoidance. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 338–345. IEEE.

[Fodor, 1975] Fodor, J. (1975). *The language of thought*. Cambridge, MA: Harvard University Press.

[Gama et al., 2021] Gama, F., Shcherban, M., Rolf, M., and Hoffmann, M. (2021). Goal-directed tactile exploration for body model learning through self-touch on a humanoid robot. *IEEE Transactions on Cognitive and Developmental Systems*.

[Graziano and Cooke, 2006] Graziano, M. and Cooke, D. (2006). Parieto-frontal interactions, personal space and defensive behavior. *Neuropsychologia*, 44:845–859.

[Haddadin et al., 2008] Haddadin, S., Albu-Schaffer, A., De Luca, A., and Hirzinger, G. (2008). Collision detection and reaction: A contribution to safe physical human-robot interaction. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3356–3363. IEEE.

[Haddadin and Croft, 2016] Haddadin, S. and Croft, E. (2016). Physical human–robot interaction. In *Springer handbook of robotics*, pages 1835–1874. Springer.

[Haddadin et al., 2017] Haddadin, S., De Luca, A., and Albu-Schäffer, A. (2017). Robot collisions: A survey on detection, isolation, and identification. *IEEE Transactions on Robotics*, 33(6):1292 – 1312.

[Haugeland, 1985] Haugeland, J. (1985). *Artificial intelligence: The very idea*. Cambridge, MA: MIT Press.

[Heed and Azañón, 2014] Heed, T. and Azañón, E. (2014). Using time to investigate space: a review of tactile temporal order judgments as a window onto spatial processing in touch. *Frontiers in psychology*, 5:76.

[Heed et al., 2015] Heed, T., Buchholz, V. N., Engel, A. K., and Röder, B. (2015). Tactile remapping: from coordinate transformation to integration in sensorimotor processing. *Trends in cognitive sciences*, 19(5):251–258.

[Hoffmann, 2017] Hoffmann, M. (2017). The role of self-touch experience in the formation of the self. In *The Development of the Self Workshop at IEEE ICDL-EpiRob 2017*.

[Hoffmann, 2021a] Hoffmann, M. (2021a). Biologically inspired robot body models and self-calibration. In Ang, M., Khatib, O., and Siciliano, B., editors, *Encyclopedia of Robotics*. Springer.

[Hoffmann, 2021b] Hoffmann, M. (2021b). Body models in humans, animals, and robots: mechanisms and plasticity. In Ataria, Y., Tanaka, S., and Gallagher, editors, *Body Schema and Body Image: New Directions*, pages 152–180. Oxford University Press.

[Hoffmann et al., 2017] Hoffmann, M., Chinn, L. K., Somogyi, E., Heed, T., Fagard, J., Lockman, J. J., and O'Regan, J. K. (2017). Development of reaching to the body in early infancy: From experiments to robotic models. In *Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 112–119.

[Hoffmann et al., 2010] Hoffmann, M., Marques, H., Hernandez Arieta, A., Sumioka, H., Lungarella, M., and Pfeifer, R. (2010). Body schema in robotics: A review. *Autonomous Mental Development, IEEE Transactions on*, 2(4):304–324.

[Hoffmann and Pfeifer, 2018] Hoffmann, M. and Pfeifer, R. (2018). Robots as powerful allies for the study of embodied cognition from the bottom up. In Newen, A., de Bruin, L., and Gallagher, S., editors, *The Oxford Handbook 4e Cognition*, chapter 45, pages 841–862. Oxford University Press.

[Hoffmann et al., 2018] Hoffmann, M., Straka, Z., Farkas, I., Vavrecka, M., and Metta, G. (2018). Robotic homunculus: Learning of artificial skin representation in a humanoid robot motivated by primary somatosensory cortex. *IEEE Transactions on Cognitive and Developmental Systems*, 10(2):163–176.

[Hollerbach et al., 2016] Hollerbach, J., Khalil, W., and Gautier, M. (2016). Model identification. In Siciliano, B. and Khatib, O., editors, *Springer Handbook of Robotics*, pages 113–138. Springer, 2nd edition.

[ISO 10218, 2011] ISO 10218 (2011). ISO 10218 Robots and robotic devices – Safety requirements for industrial robots. Standard, International Organization for Standardization, Geneva, CH.

[ISO 13855, 2010] ISO 13855 (2010). ISO 13855:2010 Safety of machinery – Positioning of safeguards with respect to the approach speeds of parts of the human body. Standard, International Organization for Standardization, Geneva, CH.

[ISO/TS 15066, 2016] ISO/TS 15066 (2016). ISO/TS 15066 Robots and robotic devices – Collaborative robots. Standard, International Organization for Standardization, Geneva, CH.

[Jacquey et al., 2020] Jacquey, L., Fagard, J., O'Regan, K., and Esseily, R. (2020). Development of body know-how during the infant's first year of life. *Enfance*, (2):175–192.

[Jiménez et al., 2001] Jiménez, P., Thomas, F., and Torras, C. (2001). 3D collision detection: a survey. *Computers & Graphics*, 25(2):269–285.

[Lacevic and Rocco, 2010] Lacevic, B. and Rocco, P. (2010). Kinetostatic danger field-a novel safety assessment for human-robot interaction. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2169–2174. IEEE.

[Lanillos et al., 2017] Lanillos, P., Dean-Leon, E., and Cheng, G. (2017). Enactive self: a study of engineering perspectives to obtain the sensorimotor self through enaction. In *2017 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 72–78. IEEE.

[Leed et al., 2019] Leed, J. E., Chinn, L. K., and Lockman, J. J. (2019). Reaching to the self: The development of infants' ability to localize targets on the body. *Psychological science*, 30(7):1063–1073.

[Magnanimo et al., 2016] Magnanimo, V., Walther, S., Tecchia, L., Natale, C., and Guhl, T. (2016). Safeguarding a mobile manipulator using dynamic safety fields. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2972–2977. IEEE.

[Maravita and Iriki, 2004] Maravita, A. and Iriki, A. (2004). Tools for the body (schema). *Trends Cogn Sci*, 8(2):79–86.

[Metta et al., 2010] Metta, G., Natale, L., Nori, F., Sandini, G., Vernon, D., Fadiga, L., von Hofsten, C., Rosander, K., Lopes, M., Santos-Victor, J., Bernardino, A., and Montesano, L. (2010). The iCub humanoid robot: An open-systems platform for research in cognitive development. *Neural Networks*, 23(8-9):1125–1134.

[Mori and Kuniyoshi, 2010] Mori, H. and Kuniyoshi, Y. (2010). A human fetus development simulation: Self-organization of behaviors through tactile sensation. In *Development and Learning (ICDL), 2010 IEEE 9th International Conference on*, pages 82–87. IEEE.

[Myowa-Yamakoshi and Takeshita, 2006] Myowa-Yamakoshi, M. and Takeshita, H. (2006). Do human fetuses anticipate self-oriented actions? a study by four-dimensional (4d) ultrasonography. *Infancy*, 10(3):289–301.

[Neisser, 1988] Neisser, U. (1988). Five kinds of self-knowledge. *Philosophical psychology*, 1(1):35–59.

[Nguyen et al., 2021] Nguyen, P. D., Georgie, Y. K., Kayhan, E., Eppe, M., Hafner, V. V., and Wermter, S. (2021). Sensorimotor representation learning for an "active self" in robots: a model survey. *KI-Künstliche Intelligenz*, 35(1):9–35.

[Nguyen et al., 2019] Nguyen, P. D., Hoffmann, M., Pattacini, U., and Metta, G. (2019). Reaching development through visuo-proprioceptive-tactile integration on a humanoid robot – a deep learning approach. In *Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 163–170.

[O'Regan and Noe, 2001] O'Regan, J. K. and Noe, A. (2001). A sensorimotor account of vision and visual consciousness. *Behavioral and Brain Sciences*, 24:939–1031.

[Orioli et al., 2018] Orioli, G., Bremner, A. J., and Farroni, T. (2018). Multisensory perception of looming and receding objects in human newborns. *Current Biology*, 28(22):R1294–R1295.

[Oudeyer and Kaplan, 2007] Oudeyer, P.-Y. and Kaplan, F. (2007). What is intrinsic motivation? A typology of computational approaches. *Frontiers in neurorobotics*, 1:6.

[Pezzulo et al., 2011] Pezzulo, G., Barsalou, L. W., Cangelosi, A., Fischer, M. H., McRae, K., and Spivey, M. J. (2011). The mechanics of embodiment: a dialog on embodiment and computational modeling. *Frontiers in psychology*, 2:5.

[Pfeifer and Bongard, 2007] Pfeifer, R. and Bongard, J. C. (2007). *How the body shapes the way we think: a new view of intelligence*. MIT Press, Cambridge, MA.

[Pfeifer and Scheier, 2001] Pfeifer, R. and Scheier, C. (2001). *Understanding intelligence*. MIT Press Cambridge, MA, USA.

[Piaget, 1952] Piaget, J. (1952). *The origins of intelligence in children*. International University Press New York.

[Piontelli, 2015] Piontelli, A. (2015). *Development of Normal Fetal Movements; The Last 15 Weeks of Gestation*. Italy: Springer-Verlag.

[Polverini et al., 2017] Polverini, M. P., Zanchettin, A. M., and Rocco, P. (2017). A computationally efficient safety assessment for collaborative robotics applications. *Robotics and Computer-Integrated Manufacturing*, 46:25–37.

[Prescott and Camilleri, 2019] Prescott, T. J. and Camilleri, D. (2019). The synthetic psychology of the self. In *Cognitive architectures*, pages 85–104. Springer.

[Pylyshyn, 1984] Pylyshyn, Z. (1984). *Computation and cognition: Toward a foundation for cognitive science*. Cambridge, MA: MIT Press.

[Rochat, 1993] Rochat, P. (1993). Hand-mouth coordination in the newborn: Morphology, determinants, and early development of a basic act. *Advances in psychology*, 97:265–288.

[Rochat, 1998] Rochat, P. (1998). Self-perception and action in infancy. *Experimental brain research*, 123(1-2):102–109.

[Roncone et al., 2016] Roncone, A., Hoffmann, M., Pattacini, U., Fadiga, L., and Metta, G. (2016). Peripersonal space and margin of safety around the body: learning tactile-visual associations in a humanoid robot with artificial skin. *PLoS ONE*, 11(10):e0163713.

[Roncone et al., 2014] Roncone, A., Hoffmann, M., Pattacini, U., and Metta, G. (2014). Automatic kinematic chain calibration using artificial skin: self-touch in the icub humanoid robot. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 2305–2312.

[Roncone et al., 2015] Roncone, A., Hoffmann, M., Pattacini, U., and Metta, G. (2015). Learning peripersonal space representation through artificial skin for avoidance and reaching with whole body surface. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 3366–3373.

[Rozlivek et al., 2021] Rozlivek, J., Rustler, L., Stepanova, K., and Hoffmann, M. (2021). Multisensorial robot calibration framework and toolbox. In *Humanoid Robots (Humanoids), IEEE-RAS International Conference on*, pages 459–466.

[Rustler et al., 2021] Rustler, L., Potocna, B., Polic, M., Stepanova, K., and Hoffmann, M. (2021). Spatial calibration of whole-body artificial skin on a humanoid robot: comparing self-contact, 3d reconstruction, and cad-based calibration. In *Humanoid Robots (Humanoids), IEEE-RAS International Conference on*, pages 445–452.

[Schillaci et al., 2016] Schillaci, G., Hafner, V. V., and Lara, B. (2016). Exploration behaviors, body representations, and simulation processes for the development of cognition in artificial agents. *Frontiers in Robotics and AI*, 3:39.

[Schmidhuber, 1991] Schmidhuber, J. (1991). A possibility for implementing curiosity and boredom in model-building neural controllers. In *Proc. of the International Conference on Simulation of Adaptive Behavior: From Animals to Animats*, pages 222–227.

[Somogyi et al., 2018] Somogyi, E., Jacquey, L., Heed, T., Hoffmann, M., Lockman, J., Granjon, L., Fagard, J., and O'Regan, J. K. (2018). Which limb is it? responses to vibrotactile stimulation in early infancy. *British Journal of Developmental Psychology*, 36(3):384–401.

[Stepanova et al., 2019] Stepanova, K., Pajdla, T., and Hoffmann, M. (2019). Robot self-calibration using multiple kinematic chains – a simulation study on the iCub humanoid robot. *IEEE Robotics and Automation Letters*, 4(2):1900–1907.

[Stepanova et al., 2022] Stepanova, K., Rozlivek, J., Puciow, F., Krsek, P., Pajdla, T., and Hoffmann, M. (2022). Automatic self-contained calibration of an industrial dual-arm robot with cameras using self-contact, planar constraints, and self-observation. *Robotics and Computer-Integrated Manufacturing*, 73:102250.

[Straka and Hoffmann, 2017] Straka, Z. and Hoffmann, M. (2017). Learning a peripersonal space representation as a visuo-tactile prediction task. In Lintas, A., Rovetta, S., Verschure, P. F., and Villa, A. E., editors, *Artificial Neural Networks and Machine Learning – ICANN 2017: 26th International Conference on Artificial Neural Networks, Alghero, Italy, September 11-14, 2017, Proceedings, Part I*, pages 101–109, Cham. Springer International Publishing.

[Svarny et al., 2021] Svarny, P., Rozlivek, J., Rustler, L., and Hoffmann, M. (2021). 3d collision-force-map for safe human-robot collaboration. *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3829–3835.

[Svarny et al., 2019] Svarny, P., Tesar, M., Behrens, J. K., and Hoffmann, M. (2019). Safe physical hri: Toward a unified treatment of speed and separation monitoring together with power and force limiting. In *Intelligent Robots and Systems (IROS), 2019 IEEE/RSJ International Conference on*, pages 7574–7581. IEEE.

[Tau and Peterson, 2010] Tau, G. Z. and Peterson, B. S. (2010). Normal development of brain circuits. *Neuropsychopharmacology*, 35(1):147.

[Thelen and Smith, 1994] Thelen, E. and Smith, L. (1994). *A Dynamic systems approach to the development of cognition and action*. MIT Press.

[Thomas et al., 2015] Thomas, B. L., Karl, J. M., and Whishaw, I. Q. (2015). Independent development of the reach and the grasp in spontaneous self-touching by human infants in the first 6 months. *Frontiers in psychology*, 5:1526.

[Van der Meer et al., 1995] Van der Meer, A., Van der Weel, F., and Lee, D. N. (1995). The functional significance of arm movements in neonates. *Science*, 267(5198):693–695.

[van der Smagt et al., 2016] van der Smagt, P., Arbib, M. A., and Metta, G. (2016). Neurorobotics: From vision to action. In *Springer handbook of robotics*, pages 2069–2094. Springer.

[Varela et al., 1991] Varela, F., Thompson, E., and E., R. (1991). *The embodied mind: cognitive science and human experience*. Cambridge, MA: MIT Press.

[Walter, 1953] Walter, G. W. (1953). *The living brain*. Norton & Co., New York.

[Yamada et al., 2016] Yamada, Y., Kanazawa, H., Iwasaki, S., Tsukahara, Y., Iwata, O., Yamada, S., and Kuniyoshi, Y. (2016). An embodied brain model of the human foetus. *Scientific Reports*, 6.

[Zanchettin et al., 2015] Zanchettin, A. M., Ceriani, N. M., Rocco, P., Ding, H., and Matthias, B. (2015). Safety in human-robot collaborative manufacturing environments: Metrics and control. *IEEE Transactions on Automation Science and Engineering*, 13(2):882–893.

[Zardykhan et al., 2019] Zardykhan, D., Svarny, P., Hoffmann, M., Shahriari, E., and Haddadin, S. (2019). Collision preventing phase-progress control for velocity adaptation in human-robot collaboration. In *Humanoid Robots (Humanoids), 2019 IEEE-RAS 18th International Conference on*. IEEE.

[Zoia et al., 2007] Zoia, S., Blason, L., D'Ottavio, G., Bulgheroni, M., Pezzetta, E., Scabar, A., and Castiello, U. (2007). Evidence of early development of action planning in the human foetus: a kinematic study. *Experimental Brain Research*, 176(2):217–226.

# Appendix A

# Robots as Powerful Allies for the Study of Embodied Cognition from the Bottom Up

Hoffmann, M. and Pfeifer, R. (2018). Robots as powerful allies for the study of embodied cognition from the bottom up. In Newen, A., de Bruin, L., and Gallagher, S., editors, *The Oxford Handbook 4e Cognition*, chapter 45, pages 841–862. Oxford University Press.

DOI: https://doi.org/10.1093/oxfordhb/9780198735410.013.45

Author contributions: The contribution of M. Hoffmann was 70%.

CHAPTER 45

# ROBOTS AS POWERFUL ALLIES FOR THE STUDY OF EMBODIED COGNITION FROM THE BOTTOM UP

MATEJ HOFFMANN AND ROLF PFEIFER

## Introduction

The study of human cognition—and human intelligence—has a long history and has kept scientists from various disciplines—philosophy, psychology, linguistics, neuroscience, artificial intelligence, and robotics—busy for many years. While there is no agreement on its definition, there is wide consensus that it is a highly complex subject matter that will require, depending on the particular position or stance, a multiplicity of methods for its investigation. Whereas, for example, psychology and neuroscience favor empirical studies on humans, artificial intelligence has proposed computational approaches, viewing cognition as information processing, as algorithms over representations. Over the last few decades, overwhelming evidence has been accumulated showing that the pure computational view is severely limited and that it must be extended to incorporate embodiment, i.e., the agent's somatic setup and its interaction with the real world, and, because they are real physical systems, robots became the tools of choice to study cognition. There have been a plethora of pertinent studies, but they all have their own intrinsic limitations. In this chapter, we demonstrate that a robotic approach, combined with information theory and a developmental perspective, promises insights into the nature of cognition that would be hard to obtain otherwise.

We start by introducing "low-level" behaviors that function without control in the traditional sense; we then move to sensorimotor processes that incorporate reflex-based loops (involving neural processing). We discuss "minimal cognition" and show how the role of embodiment can be quantified using information theory, and we introduce the so-called SMCs, or sensorimotor contingencies, which can be viewed as the very basic building blocks of cognition. Finally, we expand on how humanoid robots can be productively exploited to make inroads in the study of human cognition.

## Behavior Through Interaction

What cognitive scientists are regularly forgetting is that complex coordinated behaviors—for example, walking, running over uneven terrain, swimming, avoiding obstacles—can often be realized with no or minimal involvement of cognition/representation/computation. This is possible because of the properties of the body and the interaction with the environment, that is, the embodied and embedded nature of the agent. Robotics is well suited for providing existence proofs of this kind and then to further analyze these phenomena. We will only briefly present some of the most notable case studies.

### Low-Level Behavior: Mechanical Feedback Loops

A classical illustration of behavior in complete absence of a "brain" is the passive dynamic walker (McGeer 1990): a minimal robot that can walk without any sensors, motors, or control electronics. It loosely resembles a human, with two legs, no torso, and two arms attached to the "hips," but its ability to walk is exclusively due to the downward slope of the incline on which it walks and the mechanical parameters of the walker (mainly leg segment lengths, mass distribution, foot shape, and frictional characteristics). The walking movement is entirely the result of finely tuned mechanics on the right kind of surface. A motivation for this research is also to show how human walking is possible with minimal energy use and only limited central control. However, most of the problems that animals or robots are faced with in the real world cannot be solved solely by passive interaction of the physical body with the environment. Typically, active involvement by means of muscles/motors is required. Furthermore, the actuation pattern needs to be specified by the agent,[1] and hence a controller of some sort is required. Nevertheless, it turns out that if the physical interaction of the body with the environment is exploited, the control program can be very simple. For example, the passive dynamic walker can be modified by adding a couple of actuators and sensors and a reflex-based controller, resulting in the expansion of its niche to level ground while keeping the control effort and energy expenditure to a minimum (Collins et al. 2005).

However, in the real world, the ground is often not level and frequent corrective action needs to be taken. It turns out that often the very same mechanical system can

[1] In this chapter, we will use "agent" to describe humans, animals, or robots.

generate this corrective response. This phenomenon is known as *self-stabilization* and is a result of a mechanical feedback loop. To use dynamical systems terminology, certain trajectories (such as walking with a particular gait) have attracting properties and small perturbations are automatically corrected, without control—or one could say that "control" is inherent in the mechanical system.[2] Blickhan et al. (2007) review self-stabilizing properties of biological muscles in a paper entitled "Intelligence by Mechanics"; Koditschek et al. (2004) analyze walking insects and derive inspiration for the design of a hexapod robot with unprecedented mobility (RHex—e.g., Saranli et al. 2001).

## Sensorimotor Intelligence

Mechanical feedback loops constitute the most basic illustration of the contribution of embodiment and embeddedness to behavior. The immediate next level can be probably attributed to direct, reflex-like, sensorimotor loops. Again, robots can serve to study the mechanisms of "reactive" intelligence. Grey Walter (Walter 1953), the pioneer of this approach, built electronic machines with a minimal "brain" that displayed phototactic-like behavior. This was picked up by Valentino Braitenberg (Braitenberg 1986) who designed a whole series of two-wheeled vehicles of increasing complexity. Even the most primitive ones, in which sensors are directly connected to motors (exciting or inhibiting them), display sophisticated behaviors. Although the driving mechanisms are simple and entirely deterministic, the interaction with the real world, which brings in noise, gives rise to complex behavioral patterns that are hard to predict.

This line was picked up by Rodney Brooks, who added an explicit anti-representationalist perspective in response to the in-the-meantime-firmly-established cognitivistic paradigm (e.g., Fodor 1975; Pylyshyn 1984) and "good old-fashioned artificial intelligence" (GOFAI) (Haugeland 1985). Brooks openly attacked the GOFAI position in the seminal articles "Intelligence without Reason" (Brooks 1991a) and "Intelligence without Representation" (Brooks 1991b), and proposed *behavior-based robotics* instead. Through building robots that interact with the real world, such as insect robots (Brooks 1989), he realized that "when we examine very simple level intelligence we find that explicit representations and models of the world simply get in the way. It turns out to be better to use the world as its own model" (Brooks 1991b). Inspired by biological evolution, Brooks created a decentralized control architecture consisting of different layers; every layer is a more or less simple coupling of sensors to motors. The levels operate in parallel but are built in a hierarchy (hence the term *subsumption architecture*; Brooks 1986). The individual modules in the architecture may have internal states (the agents are thus not purely reactive any more); however, Brooks argued against calling the internal states representations (Brooks 1991b).

[2] The description is idealized—in reality, a walking machine would fall into the category of "hybrid dynamical systems," where the notions of attractivity and stability are more complicated.

## Minimal Embodied Cognition

In the case studies described in the previous section, the agents were either mere physical machines or they relied on simple direct sensorimotor loops only—resembling reflex arcs of the biological realm. They were reactive agents constrained to the "here-and-now" time scale, with no capacity for learning from experience and also no possibility of predicting the future course of events. Although remarkable behaviors were sometimes demonstrated, there are intrinsic limitations.

The introduction of first instances of internal simulation, which goes beyond the "here-and-now" time scale, is considered the hallmark of cognition by some (e.g., Clark and Grush 1999). This could be a simple forward model (as present already in insects—see Webb 2004) that provides the prediction of a future sensory state given the current state and a motor command (efference copy). Forward models could provide a possible explanation of the evolutionary origin of first simulation/emulation circuitry[3] and of environmentally decoupled thought—the agent employing primitive "models" before or instead of directly operating on the world.

> Early emulating agents would then constitute the most minimal case of what Dennett calls a Popperian creature—a creature capable of some degree of off-line reasoning and hence able (in Karl Popper's memorable phrase) to "let its hypotheses die in its stead" (Dennett 1995, p. 375). (Clark and Grush 1999, p. 7)

Importantly, we are still far from any abstract models or symbolic reasoning. Instead, we are dealing with the sensorimotor space and the possibility for the agent to extract regularities in it and later exploit this experience in accordance with its goals. For example, the agent can learn that given a certain visual stimulation, say, from a cup, a particular motor action (reach and grasp) will lead to a pattern of sensory stimulation (in humans: we can feel the cup in the hand). The sensorimotor space plays a key part here and it is critically shaped by the embodiment of the agent and its embedding in the environment: a specific motor signal only leads to a distinct result if embedded in the proper physical setup. If you change the shape and muscles of the arm, the motor signal will not result in a successful grasp.

### Quantifying the Effect of Embodiment Using Information Theory

For cognitive development of an agent, the "quality" of the sensorimotor space determines what can be learned. First, the type of sensory receptors—their mechanism

[3] See Grush (2004) for the similarities and differences between emulation theory (Grush 2004) and simulation theory (Jeannerod 2001).

of transduction—determines what kind of signals the agent's brain or controller will be receiving from the environment. Furthermore, the shape and placement of these sensors will perform an additional transformation of the information that is available in the environment.

For example, different species of insects have evolved different non-homogeneous arrangements of the light-sensitive cells in their eyes, providing an advantageous nonlinear transformation of the input for a particular task. One example is exploiting egomotion together with motion parallax to gauge distance to objects in the environment and eventually facilitate obstacle avoidance. Using a robot modeled after the facet eye of a housefly, Franceschini et al. (1992) showed that the nonlinear arrangement of the facets—more dense in the front than on the side—compensates for the motion parallax and allows uniform motion detection circuitry to be used in the entire eye, which makes it easy for the robot to avoid obstacles with little computation. These findings were confirmed in experiments with artificial evolution on real robots (Lichtensteiger 2004). Artificial eyes with designs inspired by arthropods include Song et al. (2013) and Floreano et al. (2013).

It is not always possible to pinpoint the specific transformation of sensory signals that is facilitated by the morphology as in the previous case. A more general tool is provided by the methods of information theory. Information is used in the Shannon sense here—to quantify statistical patterns in observed variables. The structure or amount of information induced by particular sensor morphology could be captured by different measures, for example, entropy. However, information (structure) in the sensory variables tells only half of the story (a "passive perception" one in this case), because organisms interact with their environments in a closed-loop fashion: sensory inputs are transformed into motor outputs, which in turn determine what is sensed next. Therefore, the "raw material" for cognition is constituted by the sensorimotor variables and it is thus crucial to study relationships between sensors and motors, as illustrated by the sensorimotor contingencies (see next section). Furthermore, time is no less important a variable. Lungarella and Sporns (2006) provide an excellent example of the use of information theoretic measures in this context. In a series of experiments with a movable camera system, they could show that, for example, the entropy in the visual field is decreased if the camera is tracking a moving visual target (a red ball) compared to the condition where the movement of the ball and the camera were uncorrelated. This is intuitively plausible, because if the object is kept in the center of the visual field, there is more "order," i.e., less entropy. A collection of case studies on information-theoretic implications of embodiment in locomotion, grasping, and visual perception is presented by Hoffmann and Pfeifer (2011).

## Sensorimotor Contingencies

Sensorimotor contingencies (SMCs) were originally presented in the influential article by O'Regan and Noë (2001) as the structure of the rules governing sensory changes produced by various motor actions. The SMCs, according to O'Regan and Noë, are the key “raw material” upon which perception, cognition, and eventually consciousness operates. Furthermore, they sketch a possible hierarchy ranging from modality-related (or apparatus-related) SMCs to object-related SMCs. The former, the modality-related SMCs, would capture the immediate effect that certain actions (or movements) have on sensory stimulation. Clearly, these would be sensory modality specific (e.g., head movement will induce a different change in the SMCs of the visual and auditory modalities—turning the head will change the visual stimulation almost entirely, whereas changes in the acoustic system will be minimal) and would strongly depend on the sensory morphology. Therefore, this concept is strongly related to what we have discussed in the previous sections: (1) different sensory morphology importantly affects the information flow induced in the sensory receptors and hence also the corresponding SMCs; (2) the effect of action is already constitutively included in the SMC notion itself.

Although conceptually very powerful, the notion of SMCs was not articulated concretely enough in O’Regan and Noë (2001) such that it could be expressed mathematically or directly transferred into a robot implementation, for example. Bührmann et al. (2013) have proposed a formal dynamical systems account of SMCs. They devised a dynamical system description for the environment and the agent, which is in turn split into body, internal state (such as neural activity), motor, and sensory dynamics. Bührmann et al. are making a distinction between sensorimotor (SM) environment, SM habitat, SM coordination, and SM strategy. The SM environment is the relation between motor actions and changes in sensory states, independent of the agent’s internal (neural) dynamics. The other notions—from SM habitat to SM strategies—add internal dynamics to the picture. SM habitat refers to trajectories in the sensorimotor space, but subject to constraints given by the internal dynamics that are responsible for generating motor commands, which may depend on previous sensory states as well—an example of closed-loop control. SM coordination then further reduces the set of possible SM trajectories to those “that contribute functionally to a task.” For example, specific patterns of squeezing an object in order to assess its hardness would be SM coordination patterns serving object discrimination. Finally, SM strategies take, in addition, “reward” or “value” for the agent into account.

As wonderfully illustrated by Beer and Williams (2015), the dynamical systems and information theory are two complementary mathematical lenses through which brain–body–environment systems can be studied. While acknowledging the merits of both frameworks as “intuition, theory, and experimental pumps” (Beer and Williams 2015), it is probably fair to say that compared to dynamical systems, information theory has been thus far more successfully applied to the analysis of real systems of higher dimensionality. This is true for both natural systems—in particular, brains (Garofalo et al. 2009; Quiroga and Panzeri 2009)—and artificial systems. Thus, to study sensorimotor contingencies in a real robot beyond the simple simulated agents of Bührmann et al. (2013) and Beer and Williams (2015), we chose to use the lens of information theory. Following up on related studies of e.g., Olsson et al. (2004), we conducted a series of studies in a real quadrupedal robot with rich nonlinear dynamics and a collection of sensors from different modalities (Hoffmann et al. 2012; Hoffmann et al. 2014; Schmidt et al. 2013) (see Box 45.1). We have applied the notion of “transfer entropy”

## Box 45.1 Sensorimotor contingencies in a quadruped robot

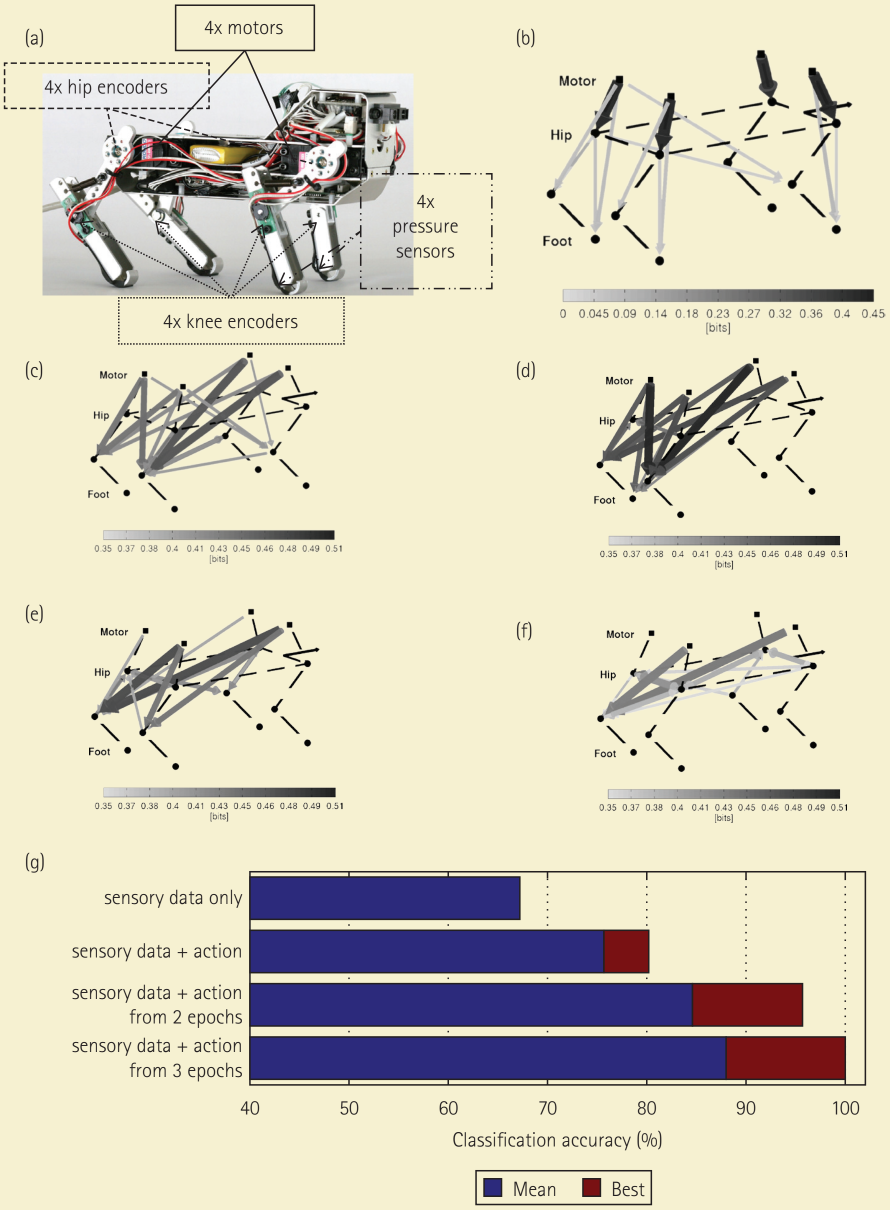


**FIGURE 45.1.** Robot "Puppy" and sensorimotor contingencies.

Experiments were conducted on the quadrupedal robot Puppy (Figure 45.1a), which has four servomotors in the hips together with encoders measuring the angle at the joint, four encoders in the passive compliant knees, and four pressure sensors on the feet. We used the notion of "transfer entropy" from information theory, which can be used to measure directed information flows between time series. In our case, the time series were collected from individual motor and sensory channels and the information transfer was calculated for every pair of channels two times, once in every direction (say, from hind right motor to front right knee encoder and also in the opposite direction). Loosely speaking, transfer entropy from channel A to channel B measures how well the future state of channel B can be predicted knowing the current state of channel A (see Schmidt et al. 2013 for details).

First, we wanted to investigate the "sensorimotor structure," i.e., the relative strengths of relationships between different sensors and motors, which is intrinsic to the robot's embodiment (body + sensor morphology only). To this end, random motor commands were applied and the relationships between motor and sensory variables were studied, closely resembling the notion of SM environment (Bührmann et al. 2013). The strongest information flows between pairs of channels were extracted and are shown overlaid over the schematic of the Puppy robot (dashed lines) in panel B. The transfer entropy is encoded as thickness and gray level of the arrows. The strongest flow occurs from the motor signals to their respective hip joint angles, which is clear because the motors directly drive the respective hip joints. The motors have a smaller influence on the knee angles (stronger in the hind legs) and on the feet pressure sensors—on the respective legs where the motor is mounted, thus illustrating that body topology was successfully extracted (at the same time, the flows from the hind leg motors and hips to the front knees highlight that the functional relationships are different than the static body structure; see also Schatz and Oudeyer 2009). These patterns are analogous to the modality-related SMCs; just as we can predict what will be the sensory changes induced by moving the head, the robot can predict the effects of moving the hind leg, say.

In a second step, we studied the relationships in the sensorimotor space when the robot was running with specific coordinated periodic movement patterns or gaits. The results for two selected gaits—turn left and bound right*—are shown in panels C and D, respectively. The flows from motors to the hip joints, which would again dominate, were left out of the visualization. The plots clearly demonstrate the important effect of specific action patterns in two ways. First, they markedly differ from the random motor command situation: the dominant flows are different and, in addition, the magnitude of the information flows is bigger (the number of bits—note the different range of the color bar compared to B), illustrating how much information structure is induced by the "neural pattern generator." Second, they also significantly differ between themselves. The "turn left" gait in panel C reveals the dominant action of the right leg and in particular the knee joint. In the "bound right" gait in D, the motor signals are predictive of the sensory stimulation in the hind knees and also the left foot. The gaits were obtained by optimizing the robot's performance for speed or for turning and thus correspond to patterns that are functionally relevant for the robot and can even be said to carry "value." Thus, in the perspective of Bührmann et al. (2013), our findings about the sensorimotor space using the gaits can be interpreted as studying the SM coordination or even SM strategy of the quadruped robot.

Finally, next to the embodiment or morphology (shape of the body and limbs, type and placement of sensors and effectors, etc.) and the brain (the neural dynamics responsible for generating the coordinated motor command sequences), the SMCs are co-determined by the environment as well. All the results thus far came from sensorimotor data collected from the robot running on a plastic foil ground (low friction). Panels E and F depict how the information flows for the bound right gait are modulated when the robot runs on a different ground (E—Styrofoam, F—rubber). The overall pattern is similar to D, but the flows to the left foot disappear, and eventually flows to the left knee joint become dominant. This is because the posture of the robot changed: the left foot contacts the ground at a different angle now, inducing less stimulation in the pressure sensor. Also, as the friction increases (from the foil over Styrofoam to rubber), the push-off during stance of the left hind leg becomes stronger, resulting in more pronounced bending of the knee. Finally, since the high-friction ground poses more resistance to the robot's movements, the trajectories are less smooth and the overall information flow drops.

While all the components (body, brain, environment) have a profound effect on the overall sensorimotor space, our analysis reveals that in this case, the gait used (as prescribed primarily by the "neural/brain" dynamics) is a more important factor than the environment (the ground)—the latter seems to modulate the basic structure of information flows induced by the gait. This has important consequences for the agent when it is to learn something about its environment and perform perceptual categorization, for example. In order to investigate this quantitatively, we have presented the robot with a terrain (the surface/ground it was running on) classification task. Relying on sensory information alone leads to significantly worse terrain classification results than when the gait is explicitly taken into account in the classification process (Hoffmann, Stepanova, and Reinstein 2014). Furthermore, in line with the predictions of the sensorimotor contingency theory, longer sensorimotor sequences are necessary for object perception (Maye and Engel 2012). That is, while in short sequences (motor command, sensory consequence), modality-related SMCs (panel B) will be dominant, longer interactions will allow objects the agent is interacting with to stand out. Using data from our robot, this is convincingly demonstrated in panel G. The first row shows classification results when using data from one sensory epoch (two seconds of locomotion) collapsed across all gaits, i.e., without the action context. Subsequent rows report results where classification was performed separately for each gait and increasingly longer interaction histories were available. "Mean" values represent the mean performance; "best" are classification results from the gait that facilitated perception the most (see Hoffmann et al. 2012 for details).

* "Turn left" was a movement pattern dominated by the action of the right hind leg that was pushing the robot forward and left. Regarding "bound right," bounding gait is a running gait used by small mammals. It is similar to gallop, and features a flight phase, but is characterized by synchronous action of every pair of legs. However, in this study, we used lower speeds without an aerial phase. In addition, the symmetry of the motor signals was slightly disrupted, resulting in a right-turning motion.

from information theory, which can be used to characterize sensorimotor flows in the robot—for example, how strongly sensors are affected by motor commands—and we tried to isolate the effects of the body, motor programs (gaits), and environment in the agent's sensorimotor space. Finally, we tested the predictions of SMC theory regarding object discrimination. In our investigations, we have chosen the situated perspective—analyzing only the relationships between sensory and motor variables that would also be available to the agent itself. However, information-theoretic methods can also be productively applied to study relationships between internal and external variables, such as between sensory or neuronal states and some properties of an external object (e.g., its size, Beer and Williams 2015; or any other property that can be expressed numerically). Using this approach, one can obtain important insights into the operation and temporal evolution of categorization, for example. Performing this in the ground discrimination scenario on the quadrupedal robot constitutes our future work.

While the studies on "minimally cognitive agents" are of fundamental importance and lead to valuable insights for our understanding of intelligent behavior, the ultimate target is, of course, human cognition. Toward this end, one may want to resort to more sophisticated tools, for example, humanoid robots.

## Human-like Cognition in Robots

In the previous section, we showed how robots can be beneficial in operationalizing, formalizing, and quantifying ideas, concepts, and theories that are important for understanding cognition but that are often not articulated in sufficient detail. An obvious implication of this analysis is that the kind of cognition that emerges will be highly dependent on the body of the agent, its sensorimotor apparatus, and the environment it is interacting with. Thus, to target human cognition, the robot's morphology—shape, type of sensors, and their distribution, materials, actuators—should resemble that of humans as closely as possible. Now we have to be realistic: approximating humans very closely would imply mimicking their physiology, the sensors in the body, and the inner organs, the muscles with comparable biological instantiation, and the bloodstream that supplies the body with energy and oxygen. Only then could the robot experience the true concept, e.g., of being thirsty or out of breath, hearing the heart pumping, blushing, or the feeling of quenching the thirst while drinking a cold beer in the summer. So, even if, on the surface, a robot might be almost indistinguishable from a human (like, for example, Hiroshi Ishiguro's recent humanoid "Erica"), we have to be aware of the fundamental differences: comparatively very few muscles and tendons, no actuators that can get sore when overused, no sensors for pain, only low-density haptic sensors, no sweat glands in the skin, and so on and so forth. Thus, "Erica" will have a very impoverished concept of drinking or feeling hot. In other words, we have to make substantial abstractions.

Just as an aside, making abstractions is nothing bad—in fact, it is one of the most crucial ingredients of any scientific explanation because it forces us to focus on the essentials, ignoring whatever is considered irrelevant (the latter most likely being the majority of things that we could potentially take into account). Thus, the specifics of the robot's cognition—its concepts, its body schema—will clearly diverge from that of humans, but the underlying principles will, at a certain level of abstraction, be the same. For example, it will have its own sensorimotor contingencies, it will form cross-modal associations through Hebbian learning, and it will explore its environment using its sensorimotor setup. So if the robot says "glass," this will relate to very different specific sensorimotor experiences, but if the robot can recognize, fill, and hand a "glass" to a human for drinking, it makes sense to say that the robot has acquired the concept of "glass."

Because the acquisition of concepts is based on sensorimotor contingencies, which in turn require actions on the part of the agent, and because the patterns of sensory stimulation are associated with the respective motor signals, the robot platforms of choice will ideally be tendon-driven—just like humans who use muscles and tendons for movements. Given our discussion on abstraction earlier, we can also study concept acquisition in robots that have motors in the joints—we just have to be aware of the concrete differences. Still, the principles governing the robot's cognition can be very similar to that of humans (see Box 45.2 for examples of different types of humanoid robots).

## BOX 45.2 Humanoid embodiment for modeling cognition

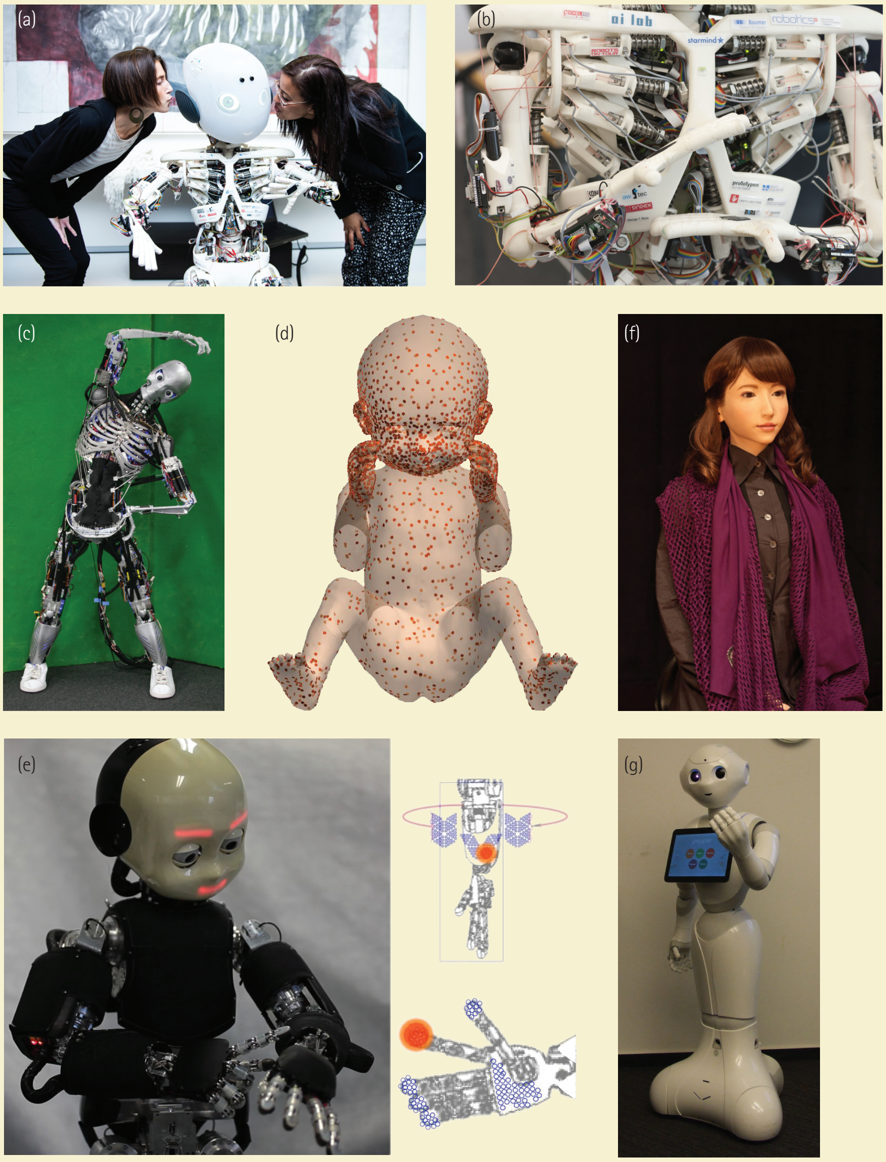


**FIGURE 45.2.** Humanoid robots.

A large number of humanoid robots have been developed over the last decades and many of them can, one way or other, be used to study human cognition. Given that all of them to date are very different from real humans—each of them, implicitly or explicitly, embodies certain types of abstractions—there is no universal platform, but they have all been developed with specific goals in mind. Here we present a few examples and discuss the ways in which they are employed in trying to ferret out the principles of human cognition. The categories shown in Figure 45.2 are musculoskeletal robots (Roboy and Kenshiro), "baby" robots with sensorized skins (iCub and fetus simulators), and social interaction robots (Erica and Pepper).

In order to use the robots for learning their own complex dynamics and for building up a body schema, both Roboy and Kenshiro (Nakanishi et al. 2012) need to be equipped with many sensors so that they can "experience" the effect of a particular actuation pattern. Given rich sensory feedback, using the principle that every action leads to sensory stimulation, both these robots can, in principle, employ motor babbling in order to learn how to move. Especially for Kenshiro, with his very large number of muscles, learning is a must. A very important step in this direction is the work of Richter et al. (2016), who have combined a musculoskeletal robotics toolkit (Myorobotics) with a scalable neuromorphic computing platform (SpiNNaker) and demonstrated control of a musculoskeletal joint with a simulated cerebellum.

Finally, if the interest is social interaction, it might be more productive to use robots like Erica or Pepper. Both Erica and Pepper are somewhat limited in their sensorimotor abilities (especially haptics), but are endowed with speech understanding and generation facilities; they can recognize faces and emotions; and they can realistically display any kind of facial expression.

**Musculoskeletal robots: Roboy and Kenshiro**

Figure 45.2a. Roboy overview: The musculoskeletal design can be clearly observed. At this point, Roboy has 48 "muscles." Eight are dedicated to each of the shoulder joints. This can no longer be sensibly programmed: learning is a necessity. Currently, Roboy serves as a research platform for the EU/FET Human Brain Project to study, among other things, the effect of brain lesions on the musculoskeletal system. Because it has the ability to express a vast spectrum of emotions, it can also be employed to investigate human–robot interaction, and as an entertainment platform.

Credit: © Embassy of Switzerland in the United States of America.

Figure 45.2b. Close-up of the muscle-tendon system. Although the shoulder joint is distinctly dissimilar to a human one—for example, it doesn't have a shoulder blade—it is controlled by eight muscles, which require substantial skills in order to move properly: which muscles have to be actuated to what extent in order to achieve a desired movement?

Credit: © Erik Tham/Corbis Documentary/Getty Images.

Figure 45.2c. Kenshiro's musculoskeletal setup. The musculoskeletal design is clearly visible. At this point, Kenshiro has 160 "muscles"—50 in the legs, 76 in the trunk, 12 in the

shoulder, and 22 in the neck. In terms of musculoskeletal system, it is the one robot that most closely resembles the human. So, if learning of the dynamics in this system is the goal, Kenshiro will be the robot of choice. Note that although Kenshiro is "closest" to a human in this respect, it is still subject to enormous abstractions. Currently, Kenshiro serves as a research platform at the University of Tokyo to investigate tendon-controlled systems with very many degrees of freedom (Nakanishi et al. 2012).

Credit: Photo courtesy Yuki Asano.

**"Baby" robots with sensitive skins**

Figure 45.2d. Fetus simulator. A musculoskeletal model of human fetus at 32 weeks of gestation has been constructed and coupled with a brain model comprising 2.6 million spiking neurons (Yamada et al. 2016). The figure shows the tactile sensor distribution, which was based on human two-point discrimination data.

Reproduced from Yasunori Yamada, Hoshinori Kanazawa, Sho Iwasaki, Yuki Tsukahara, Osuke Iwata, Shigehito Yamada, and Yasuo Kuniyoshi, An Embodied Brain Model of the Human Foetus, *Scientific Reports*, 6 (27893), Figure 1d, doi:10.1038/srep27893 © 2016 Yasunori Yamada, Hoshinori Kanazawa, Sho Iwasaki, Yuki Tsukahara, Osuke Iwata, Shigehito Yamada, and Yasuo Kuniyoshi. This work is licensed under the Creative Commons Attribution 4.0 International License (CC BY 4.0). It is attributed to the authors Yasunori Yamada, Hoshinori Kanazawa, Sho Iwasaki, Yuki Tsukahara, Osuke Iwata, Shigehito Yamada, and Yasuo Kuniyoshi.

Figure 45.2e. The iCub baby humanoid robot. The iCub (Metta et al. 2010) has the size of a roughly four-year-old child and corresponding sensorimotor capacities: 53 degrees of freedom (electrical motors), two stereo cameras in a biomimetic arrangement, and over 4,000 tactile sensors covering its body. The panel shows the robot performing self-touch and corresponding activations in the tactile arrays of the left forearm and right index finger.

**Social interaction robots: Erica and Pepper**

Figure 45.2f. Erica, the latest creation of Prof. Hiroshi Ishiguro, was designed specifically with the goal of imitating human speech and body language patterns, in order to have "highly natural" conversations. It also serves as a tool to study human–robot interaction, and social interaction in general. Moreover, because of its close resemblance to humans, the "uncanny valley"—the fact that people get uneasy when the robots are too human-like—hypothesis can be further explored and analyzed (see, e.g., Rosenthal-von der Pütten, Marieke, and Weiss 2014, where the Geminoid HI-1 modeled after Prof. Ishiguro was used).

Credit: Photo courtesy of Hiroshi Ishiguro Laboratory, ATR and Osaka University.

Figure 45.2g. Pepper, a robot developed by Aldebaran (now Softbank Robotics), although much simpler (and much cheaper!) than Erica, is used successfully on the one hand to study social interaction, for entertainment, and to perform certain tasks (such as selling Nespresso machines to customers in Japan).

## The Role of Development

A very powerful approach to deepen our understanding of cognition, and one that has been around for a long time in psychology and neuroscience, is to study ontogenetic development. During the past two decades or so, this idea has been adopted by the robotics community and has led to a thriving research field dubbed "developmental robotics." Now, a crucial part of ontogenesis takes place in the uterus. There, tactile sense is the first to develop (Bernhardt 1987) and may thus play a key role in the organism's learning about first sensorimotor contingencies, in particular, those pertaining to its own body (e.g., hand-to-mouth behaviors). Motivated by this fact, Mori and Kuniyoshi (2010) developed a musculoskeletal fetal simulator with over 1,500 tactile receptors, and studied the effect of their distribution on the emergence of sensorimotor behaviors. Importantly, with a natural (non-homogeneous) distribution, the fetus developed "normal" kicking and jerking movements (i.e., similar to those observed in a human fetus), whereas with a homogeneous allocation it did not develop any of these behaviors. Yamada et al. (2016), using a similar fetal simulator and a large spiking neural network brain model, have further studied the effects of intrauterine (vs. extrauterine) sensorimotor experiences on cortical learning of body representations. A physical version—the fetusoid—is currently under development (Mori et al. 2015). Somatosensory (tactile and proprioceptive) inputs continue to be of key importance also in early infancy when "infants engage in exploration of their own body as it moves and acts in the environment. They babble and touch their own body, attracted and actively involved in investigating the rich intermodal redundancies, temporal contingencies, and spatial congruence of self-perception" (Rochat 1998, p. 102). The iCub baby humanoid robot (Metta et al. 2010) (Box 45.2E), equipped with a whole-body tactile array (Maiolino et al. 2013) comprising over 4,000 elements, is an ideal platform to study these processes. The study of Roncone et al. (2014) on self-calibration using self-touch is a first step in this direction.

## Applications Of Human-Like Robots

Finally, this research strand—employing humanoid robots to study human cognition—has also important applications. In traditional domains and conventional tasks—such as pick-and-place operations in an industrial environment—current factory automation robots are doing just fine. However, robots are starting to leave these constrained domains, entering environments that are far less structured and are starting to share their living space with humans. As a consequence, they need to dynamically adapt to unpredictable interactions and guarantee their own as well as others' safety at every moment. In such cases, more human-like characteristics—both physical and "mental"—are desirable. Box 45.3 illustrates how more brain-like body representations can help robots to become more autonomous, robust, and safe. The possibilities for future applications of robots with cognitive capacities are enormous, especially in the rapidly

## BOX 45.3 Body schema in humans vs. robots

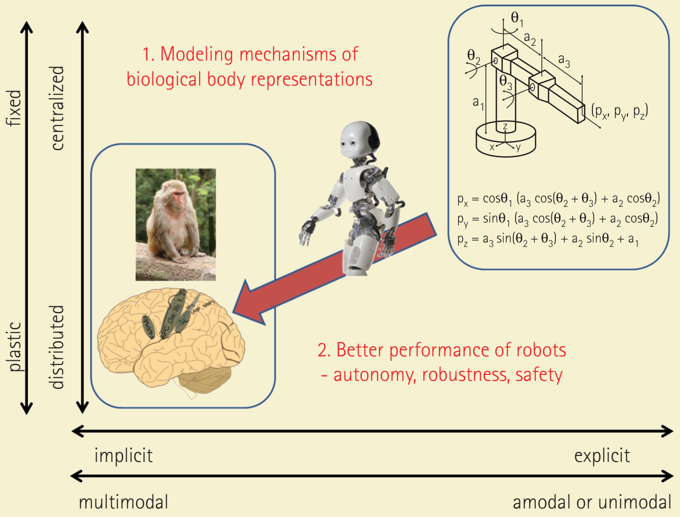

fixed
plastic
centralized
distributed
1. Modeling mechanisms of biological body representations
θ1
θ2
θ3
a1
a2
a3
(px, py, pz)
z
x
y
px = cosθ1 (a3 cos(θ2 + θ3) + a2 cosθ2)
py = sinθ1 (a3 cos(θ2 + θ3) + a2 cosθ2)
pz = a3 sin(θ2 + θ3) + a2 sinθ2 + a1
2. Better performance of robots – autonomy, robustness, safety
implicit
explicit
multimodal
amodal or unimodal

FIGURE 45.3. Characteristics of body representations.

Credit: Monkey photo source: Einar Fredriksen/Flickr/Attribution-ShareAlike 4.0 International (CC BY-SA 4.0)
Credit: Brain image source: Hugh Guiney/Attribution-ShareAlike 3.0 Unported (CC BY-SA 3.0)
Credit: Line drawing and equations source: Reproduced with the permission of Dr. Hugh Jack from http://www.engineeronadisk.com
Credit: iCub Robot source: © iCub Facility—IIT, 2017

A typical example of a traditional robot and its mathematical model is depicted in the upper right of Figure 45.3. The robot is an arm consisting of three segments with three joints between the base and the final part—the end-effector. Its model is below the robot—the forward kinematics equations that relate configuration of the robot (joint positions $\theta_1$, $\theta_2$, $\theta_3$) to the Cartesian position of the end-effector ($p_x$, $p_y$, $p_z$). The model has the following characteristics: (1) it is explicit—there is a one-to-one correspondence between its body and the model ($a_1$ in the model is the length of the first arm segment, for example); (2) it is unimodal—the equations directly describe physical reality; one sensory modality (proprioception—joint angle values) is needed to get the correct mapping in the current robot state; (3) it is centralized—there is only one model that describes the whole robot; (4) it is fixed—normally, this mapping is set and does not change during the robot operation. Other models/mappings are typically needed for robot operation, such as inverse kinematics, differential kinematics, or models of dynamics (dealing with forces and torques), but they would all share the above-mentioned characteristics (see Hoffmann et al. 2010 for a survey).

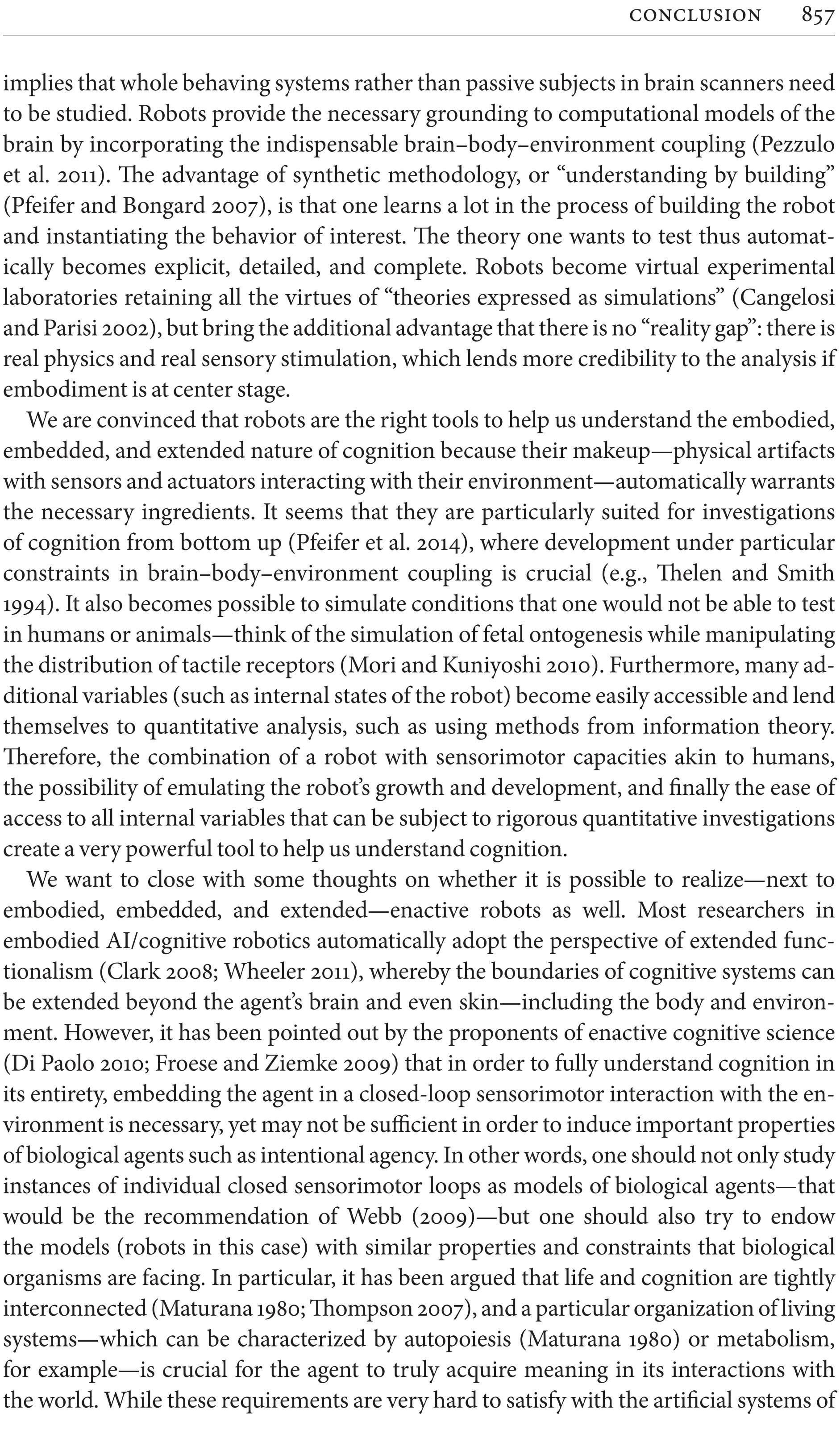
As pointed out earlier, animals and humans have different bodies than robots; they also have very different ways of representing them in their brains. The panel in the lower left shows the rhesus macaque and below some of the key areas of its brain that deal with body representations (see, e.g., Graziano and Botvinick 2002). There is ample evidence that these representations differ widely from the ones traditionally used in robotics—namely, "the body in the brain" would be (1) implicitly represented—there would hardly be a "place" or a "circuit" encoding, say, the length of a forearm; such information is most likely only indirectly available and possibly in relation to other variables; (2) multimodal—drawing mainly from somatosensory (tactile and proprioceptive) and visual, but also vestibular (inertial) and closely coupled to motor information; (3) distributed—there are numerous distinct, but partially overlapping and interacting representations that are dynamically recruited depending on context and task; (4) plastic—adapting over both long (ontogenesis) and short time scales, as adaptation to tool use (e.g., Iriki et al. 1996) or various body illusions testify (e.g., humans start feeling ownership over a rubber hand after minutes of synchronous tactile stimulations of the hand replica and their real hand under a table; Botvinick and Cohen 1998).

The iCub robot "walking" from the top right to the bottom left in the figure is illustrating two things. First, in order to be able to model the mechanisms of biological body representations, the traditional robotic models are of little use—a radically different approach needs to be taken. Second, by making the robot models more brain-like, we hope to inherit some of the desirable properties typical of how humans and animals master their highly complex bodies. Autonomy and robustness or resilience are one such case. It is not realistic to think that conditions, including the body, will stay constant over time and a model given to the robot by the manufacturer will always work. Inaccuracies will creep in due to wear and tear and possibly even more dramatic changes can occur (e.g., a joint becomes blocked). Humans and animals display a remarkable capacity for dealing with such changes: their models dynamically adapt to muscle fatigue, for example, or temporarily incorporate objects like tools after working with them, or reallocate "brain territory" to different body parts in case of amputation of a limb. Robots thus also need to perform continuous self-modeling (Bongard et al. 2006) in order to cope with such changes. Finally, unlike factory robots that blindly execute their trajectories and thus need to operate in cages, humans and animals use multimodal information to extend the representation of their bodies to the space immediately surrounding them (also called peripersonal space). They construct a "margin of safety," a virtual "bubble" around their bodies that allows them to respond to potential threats such as looming objects, warranting safety for them and also their surroundings (e.g., Graziano and Cooke 2006). This is highly desirable in robots as well, and can transform them from dangerous machines to collaborators possessing whole-body awareness like we do. First steps along these lines in the iCub were presented by Roncone et al. (2016).

growing area of service robotics, where robots perform tasks in human environments. Rather than accomplishing them autonomously, they often do it in cooperation with humans, which constitutes a big trend in the field. In cooperative tasks, it is of course crucial that the robots understand the common goals and the intentions of the humans in order to be successful. In other words, they require substantial cognitive skills. We have barely started exploiting the vast potential of these types of cognitive machines.

## Conclusion

Our analysis so far has demonstrated that robots fit squarely into the embodied and pragmatic (action-oriented) turn in cognitive sciences (e.g., Engel et al. 2013), which



implies that whole behaving systems rather than passive subjects in brain scanners need to be studied. Robots provide the necessary grounding to computational models of the brain by incorporating the indispensable brain–body–environment coupling (Pezzulo et al. 2011). The advantage of synthetic methodology, or "understanding by building" (Pfeifer and Bongard 2007), is that one learns a lot in the process of building the robot and instantiating the behavior of interest. The theory one wants to test thus automatically becomes explicit, detailed, and complete. Robots become virtual experimental laboratories retaining all the virtues of "theories expressed as simulations" (Cangelosi and Parisi 2002), but bring the additional advantage that there is no "reality gap": there is real physics and real sensory stimulation, which lends more credibility to the analysis if embodiment is at center stage.

We are convinced that robots are the right tools to help us understand the embodied, embedded, and extended nature of cognition because their makeup—physical artifacts with sensors and actuators interacting with their environment—automatically warrants the necessary ingredients. It seems that they are particularly suited for investigations of cognition from bottom up (Pfeifer et al. 2014), where development under particular constraints in brain–body–environment coupling is crucial (e.g., Thelen and Smith 1994). It also becomes possible to simulate conditions that one would not be able to test in humans or animals—think of the simulation of fetal ontogenesis while manipulating the distribution of tactile receptors (Mori and Kuniyoshi 2010). Furthermore, many additional variables (such as internal states of the robot) become easily accessible and lend themselves to quantitative analysis, such as using methods from information theory. Therefore, the combination of a robot with sensorimotor capacities akin to humans, the possibility of emulating the robot's growth and development, and finally the ease of access to all internal variables that can be subject to rigorous quantitative investigations create a very powerful tool to help us understand cognition.

We want to close with some thoughts on whether it is possible to realize—next to embodied, embedded, and extended—enactive robots as well. Most researchers in embodied AI/cognitive robotics automatically adopt the perspective of extended functionalism (Clark 2008; Wheeler 2011), whereby the boundaries of cognitive systems can be extended beyond the agent's brain and even skin—including the body and environment. However, it has been pointed out by the proponents of enactive cognitive science (Di Paolo 2010; Froese and Ziemke 2009) that in order to fully understand cognition in its entirety, embedding the agent in a closed-loop sensorimotor interaction with the environment is necessary, yet may not be sufficient in order to induce important properties of biological agents such as intentional agency. In other words, one should not only study instances of individual closed sensorimotor loops as models of biological agents—that would be the recommendation of Webb (2009)—but one should also try to endow the models (robots in this case) with similar properties and constraints that biological organisms are facing. In particular, it has been argued that life and cognition are tightly interconnected (Maturana 1980; Thompson 2007), and a particular organization of living systems—which can be characterized by autopoiesis (Maturana 1980) or metabolism, for example—is crucial for the agent to truly acquire meaning in its interactions with the world. While these requirements are very hard to satisfy with the artificial systems of today, Di Paolo (2010) proposes a way out: robots need not metabolize, but they should be subject to so-called precarious conditions. That is, the success of a particular instantiation of sensorimotor loops or neural vehicles in the agent is to be measured against some viability criterion that is intrinsic to the organization of the agent (e.g., loss of battery charge, overheating leading to electronic board problems resulting in loss of mobility, etc.). The control structure may develop over time, but the viability constraint needs to be satisfied, otherwise the agent "dies" (McFarland and Boesser 1993). In a similar vein, in order to move from embodied to enactive AI, Froese and Ziemke (2009) propose to extend the design principles for autonomous agents of Pfeifer and Scheier (2001), requiring the agents to generate their own systemic identity and regulate their sensorimotor interaction with the environment in relation to a viability constraint. The unfortunate implication, however, is that research along these lines will in the short term most likely not produce useful artifacts. On the other hand, this approach may eventually give rise to truly autonomous robots with unimaginable application potential.

## Acknowledgments

M.H. was supported by a Marie Curie Intra European Fellowship (iCub Body Schema 625727) within the 7th European Community Framework Programme and the Czech Science Foundation under Project GA17-15697Y.

## References

Beer, R.D. and Williams, P.L. (2015). Information processing and dynamics in minimally cognitive agents. *Cognitive Science*, 39, 1–38.

Bernhardt, J. (1987). Sensory capabilities of the fetus. *MCN: The American Journal of Maternal/Child Nursing*, 12(1), 44–7.

Blickhan, R., Seyfarth, A., Geyer, H., Grimmer, S., Wagner, H., and Günther, M. et al. (2007). Intelligence by mechanics. *Philosophical transactions. Series A*, 365, 199–220.

Bongard, J., Zykov, V., and Lipson, H. (2006). Resilient machines through continuous self-modeling. *Science*, 314, 1118–21.

Botvinick, M. and Cohen, J. (1998). Rubber hands "feel" touch that eyes see. *Nature*, 391(6669), 756.

Braitenberg, V. (1986). *Vehicles—experiments in synthetic psychology*. Cambridge, MA: MIT Press.

Brooks, R. (1986). A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2(1), 14–23.

Brooks, R.A. (1989). A robot that walks: emergent behaviors from a carefully evolved network. *Neural Computation*, 1, 153–62.

Brooks, R.A. (1991a). Intelligence without reason. In: J. Myopoulos (ed.), Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (vol. 1). San Francisco, USA: Morgan Kaufmann, pp. 569–95.

Brooks, R.A. (1991b). Intelligence without representation. *Artificial Intelligence*, 47, 139–59.

Bührmann, T., Di Paolo, E., and Barandiaran, X. (2013). A dynamical systems account of sensorimotor contingencies. *Frontiers in Psychology*, 4, 285.

Cangelosi, A. and Parisi, D. (2002). Computer simulation: a new scientific approach to the study of language evolution. In: *Simulating the evolution of language*. London: Springer Science & Business Media, pp. 3–28.

Clark, A. (2008). *Supersizing the mind: embodiment, action, and cognitive extension*. New York: Oxford University Press.

Clark, A. and Grush, R. (1999). Towards cognitive robotics. *Adaptive Behaviour*, 7(1), 5–16.

Collins, S., Ruina, A., Tedrake, R., and Wisse, M. (2005). Efficient bipedal robots based on passive dynamic walkers. *Science*, 307, 1082–5.

Dennett, D. (1995). *Darwin's dangerous idea*. New York: Simon & Schuster.

Di Paolo, E. (2010). Robotics inspired in the organism. *Intellectica*, 53–54, 129–62.

Engel, A.K., Maye, A., Kurthen, M., and König, P. (2013). Where's the action? The pragmatic turn in cognitive science. *Trends in Cognitive Sciences*, 17(5), 202–9.

Floreano, D., Pericet-Camara, R., Viollet, S., Ruffier, F., Brückner, A., Leitel, R. et al. (2013). Miniature curved artificial compound eyes. *Proceedings of the National Academy of Sciences*, 110(23), 9267–72.

Fodor, J. (1975). *The language of thought*. Cambridge, MA: Harvard University Press.

Franceschini, N., Pichon, J., and Blanes, C. (1992). From insect vision to robot vision. *Philosophical transactions of the Royal Society of London. Series B*, *Biological sciences*, 337, 283–94.

Froese, T. and Ziemke, T. (2009). Enactive artificial intelligence: investigating the systemic organization of life and mind. *Artificial Intelligence*, 173(3), 466–500.

Garofalo, M., Nieus, T., Massobrio, P., and Martinoia, S. (2009). Evaluation of the performance of information theory-based methods and cross-correlation to estimate the functional connectivity in cortical networks. *PLoS ONE*, 4(8), e6482.

Graziano, M. and Botvinick, M. (2002). How the brain represents the body: insights from neurophysiology and psychology. In: W. Prinz and B. Hommel (eds.), *Common mechanisms in perception and action: attention and performance*. New York: Oxford University Press, 136–57.

Graziano, M. and Cooke, D. (2006). Parieto-frontal interactions, personal space, and defensive behavior. *Neuropsychologia*, 44(6), 845–59.

Grush, R. (2004). The emulation theory of representation—motor control, imagery, and perception. *Behavioral and Brain* Sciences, 27, 377–442.

Haugeland, J. (1985). *Artificial intelligence: the very idea*. Cambridge, MA: MIT Press.

Hoffmann, M., Marques, H., Arieta, A., Sumioka, H., Lungarella, M., and Pfeifer, R. (2010). Body schema in robotics: a review. *IEEE Transactions on Autonomous Mental Development*, 2(4), 304–24.

Hoffmann, M. and Pfeifer, R. (2011). The implications of embodiment for behavior and cognition: animal and robotic case studies. In: W. Tschacher and C. Bergomi (eds.), *The implications of embodiment: cognition and communication*. Exeter: Imprint Academic, pp. 31–58.

Hoffmann, M., Schmidt, N.M., Pfeifer, R., Engel, A.K., and Maye, A. (2012). *Using sensorimotor contingencies for terrain discrimination and adaptive walking behavior in the quadruped robot Puppy*. In: T. Ziemke, C. Balkenius, and J. Hallam (eds.), *From animals to animats 12*. SAB 2012. Lecture Notes in Computer Science (vol. 7426). Berlin, Heidelberg: Springer, pp. 54–64.

Hoffmann, M., Stepanova, K., and Reinstein, M. (2014). The effect of motor action and different sensory modalities on terrain classification in a quadruped robot running with multiple gaits. *Robotics and Autonomous Systems*, 62(12), 1790–8.

Iriki, A., Tanaka, M., and Iwamura, Y. (1996). Coding of modified body schema during tool use by macaque postcentral neurones. *Neuroreport*, 7, 2325–30.

Jeannerod, M. (2001). Neural simulation of action: a unifying mechanism for motor cognition. *NeuroImage*, 14, 103–9.

Koditschek, D.E., Full, R.J., and Buehler, M. (2004). Mechanical aspects of legged locomotion control. *Arthropod Structure and Development*, 33, 251–72.

Lichtensteiger, L. (2004). *On the interdependence of morphology and control for intelligent behavior* [PhD dissertation]. Zurich: University of Zurich.

Lungarella, M. and Sporns, O. (2006). Mapping information flow in sensorimotor networks. *PLoS Computational Biology*, 2, 1301–12.

Maiolino, P., Maggiali, M., Cannata, G., Metta, G., and Natale, L. (2013). A flexible and robust large scale capacitive tactile system for robots. *Sensors Journal, IEEE*, 13(10), 3910–7.

Maturana, H.a.V.F. (1980). *Autopoiesis and cognition: the realization of the living*. Dordrecht: D. Reidel Publishing.

Maye, A. and Engel, A.K. (2012). Time scales of sensorimotor contingencies. In: H. Zhang, A. Hussain, D. Liu, and Z. Wang (eds.), *Advances in brain inspired cognitive systems*. BICS 2012. Lecture Notes in Computer Science (vol. 7366). Berlin, Heidelberg: Springer, 240–9.

McGeer, T. (1990). Passive dynamic walking. *The International Journal of Robotics Research*, 9(2), 62–82.

Metta, G., Natale, L., Noei, F., Sandini, G., Vernon, D., Fadiga L. et al. (2010). The iCub humanoid robot: an open-systems platform for research in cognitive development. *Neural Networks*, 23(8–9), 1125–34.

Mori, H., Akutsu, D., and Asada, M. (2015). Fetusoid35: a robot research platform for neural development of both fetuses and preterm infants and for developmental care. In: A. Duff, N.F. Lepora, A. Mura, T.J. Prescott, and P.F.M.J. Verschure (eds.), *Biomimetic and biohybrid systems*. Living Machines 2014. Lecture Notes in Computer Science (vol. 8608). New York: Springer International Publishing, pp. 411–13.

Mori, H. and Kuniyoshi, Y. (2010). A human fetus development simulation: self-organization of behaviors through tactile sensation. In: *2010 IEEE 9th International Conference on Development and Learning*. doi:10.1109/DEVLRN.2010.5578860

Nakanishi, Y., Asano, Y., Kozuki, T., Mizoguchi, H., Motegi, Y., Osada, M. et al. (2012). Design concept of detail musculoskeletal humanoid "Kenshiro"—toward a real human body musculoskeletal simulator. In: *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*. doi:10.1109/HUMANOIDS.2012.6651491

Olsson, L., Nehaniv, C.L., and Polani, D. (2004). Sensory channel grouping and structure from uninterpreted sensory data. In: *Proceedings. 2004 NASA/DoD Conference on Evolvable Hardware, 2004*. doi:10.1109/EH.2004.1310825

O'Regan, J.K. and Noë, A. (2001). A sensorimotor account of vision and visual consciousness. *Behavioral and Brain Sciences*, 24, 939–1031.

Pezzulo, G., Barsalou, L.W., Cangelosi, A., Fischer, M.H., McRae, K., and Spivey, M.J. (2011). The mechanics of embodiment: a dialog on embodiment and computational modeling. *Frontiers in Psychology*, 2, 5.

Pfeifer, R. and Bongard, J.C. (2007). *How the body shapes the way we think: a new view of intelligence*. Cambridge, MA: MIT Press.

Pfeifer, R., Iida, F., and Lungarella, M. (2014). Cognition from the bottom up: on biological inspiration, body morphology, and soft materials. *Trends in Congnitive Sciences*, 18(8), 404–13.

Pfeifer, R. and Scheier, C. (2001). *Understanding intelligence*. Cambridge, MA: MIT Press.

Pylyshyn, Z. (1984). *Computation and cognition: toward a foundation for cognitive science*. Cambridge, MA: MIT Press.

Quiroga, R.Q., and Panzeri, S. (2009). Extracting information from neuronal populations: information theory and decoding approaches. *Nature Reviews Neuroscience*, 10(3), 173–85.

Richter, C., Jentzsch, S., Hostettler, R., Garrido, J.A., Ros, E., Knoll, A. et al. (2016). Musculoskeletal robots: scalability in neural control. *IEEE Robotics & Automation Magazine*, 23(4), 128–37. doi:10.1109/MRA.2016.2535081

Rochat, P. (1998). Self-perception and action in infancy. *Experimental Brain Research*, 123, 102–9.

Roncone, A., Hoffmann, M., Pattacini, U., Fadiga, L., and Metta, G. (2016). Peripersonal space and margin of safety around the body: learning tactile-visual associations in a humanoid robot with artificial skin. *PLoS ONE*, 11(10), e0163713.

Roncone, A., Hoffmann, M., Pattacini, U., and Metta, G. (2014). Automatic kinematic chain calibration using artificial skin: self-touch in the iCub humanoid robot. In: *2014 IEEE International Conference on Robotics and Automation (ICRA)*. doi:10.1109/ICRA.2014.6907178

Rosenthal-von der Pütten, A.M., Marieke, A., and Weiss, A. (2014). The uncanny in the wild: analysis of unscripted human–android interaction in the field. *International Journal of Social Robotics*, 6(1), 67–83.

Saranli, U., Buehler, M., and Koditschek, D. (2001). RHex: a simple and highly mobile hexapod robot. The *International Journal of Robotics Research*, 20, 616–31.

Schatz, T. and Oudeyer, P.Y. (2009). Learning motor dependent Crutchfield's information distance to anticipate changes in the topology of sensory body maps. *2009 IEEE 8th International Conference on Development and Learning*. doi:10.1109/DEVLRN.2009.5175526

Schmidt, N., Hoffmann, M., Nakajima, K., and Pfeifer, R. (2013). Bootstrapping perception using information theory: case studies in a quadruped robot running on different grounds. *Advances in Complex Systems*, 16(2-3), 1250078.

Song, Y.M. et al. (2013). Digital cameras with designs inspired by the arthropod eye. *Nature*, 497(7447), 95–9.

Thelen, E. and Smith, L. (1994). *A dynamic systems approach to the development of cognition and action*. Cambridge, MA: MIT Press.

Thompson, E. (2007). *Mind in life: biology, phenomenology, and the sciences of mind*. Cambridge, MA: MIT Press.

Walter, G.W. (1953). *The living brain*. New York: Norton & Co.

Webb, B. (2004). Neural mechanisms for prediction: do insects have forward models? *Trends in Neurosciences*, 27(5), 278–82.

Webb, B. (2009). Animals versus animats: or why not model the real iguana? *Adaptive Behavior*, 17, 269–86.

Wheeler, M. (2011). Embodied cogntion and the extended mind. In: J. Garvey (ed.), *The Continuum companion to philosophy of mind*. London: Continuum, pp. 220–36.

Yamada, Y., Kanazawa, H., Iwasaki, S., Tsukahara, Y., Iwata, O., Yamada, S. et al. (2016). An embodied brain model of the human foetus. *Scientific Reports,* 6. doi:10.1038/srep27893

# Appendix B

# Body models in humans, animals, and robots: mechanisms and plasticity

Hoffmann, M. (2021). Body models in humans, animals, and robots: mechanisms and plasticity. In Ataria, Y., Tanaka, S., and Gallagher, S., editors, *Body Schema and Body Image: New Directions*, pages 152–180. Oxford University Press.

DOI: https://doi.org/10.1093/oso/9780198851721.003.0010

# 10

# Body models in humans, animals, and robots: mechanisms and plasticity

*Matej Hoffmann*

## 10.1 Introduction

Ulric Neisser distinguishes five different selves: the ecological self, the interpersonal self, the extended self, the private self, and the conceptual self (1988). The high-level facets of the self—accessible to consciousness, incorporating linguistic information, etc.—have been receiving relatively more attention. However, here I will focus on the lowest level—the ecological or sensorimotor self or the 'body schema'—which constitutes a key foundation for the rest. The description by Graziano and Botvinick (2002) nicely expresses the sensorimotor, multimodal, and spatial nature of the body representations on which I focus: 'implicit knowledge structure that encodes the body's form, the constraints on how the body's parts can be configured, and the consequences of this configuration on touch, vision, and movement'.

The first key question I want to address is what the fundamental differences are in which animals, humans, and robots represent their bodies. While the main goal is to get an understanding of the mechanisms of 'the body in the biological brain', the 'robot world' can be instrumental here in two ways. First, there is a large body of mature mathematical tools for representing kinematics and dynamics and for employing these representations in movement planning and control, as well as for learning models of physical systems (system identification). These constitute, in some sense, the 'ideal world', a neat mathematical description of the problem, which opens up a useful perspective on the body models that evolution has arrived at. Second, robots can serve as embodied computational models of biological body representations. Humanoid robots possess morphologies—physical characteristics, as well as sensory and motor apparatuses—that are, in some respects, akin to human bodies and can thus be used to expand the domain of computational modelling by anchoring it to the physical environment and a physical body and allowing for instantiation of complete sensorimotor loops.

The second key question is: Which properties of the biological 'body schema' could be transferred to robots to make them more adaptive and resilient? On one hand, robots are endowed with neatly engineered body models and control algorithms. Yet, in many respects, their performance in commanding their bodies in unstructured

Matej Hoffmann, *Body models in humans, animals, and robots: mechanisms and plasticity* In: *Body Schema and Body Image.*
Edited by: Yochai Ataria, Shogo Tanaka, and Shaun Gallagher, Oxford University Press. © Oxford University Press 2021.

environments, adapting to failures or tools, etc., is still hugely lagging behind their biological counterparts. Therefore, Section 10.4 will examine which of the characteristics of the 'body in the brain' robots should take on board.

## 10.2 Body models—octopus, primates, robots

Biological and artificial agents have very different bodies, as well as very different representations thereof. In this section, I will look at some of the characteristics of bodies and 'brains' of invertebrates, primates, and robots. Reaching will be used throughout as a behaviour that requires some form of—implicit or explicit—body model.

### 10.2.1 The invertebrate brain and reaching in the octopus

[1]Unlike in vertebrates, invertebrate species show an enormous diversity in body plans and nervous organization (Marder, 2007; Zullo & Hochner, 2011). With more complex bodies and nervous structures, there is a tendency toward centralization with the formation of a structured cephalic ganglion. Ganglia or their groups become larger and tend to form semi-autonomous systems for sensorimotor control. Brain development in the rostral part of the animal comes also from the presence of distal sensing such as vision. Within the higher nervous system, sensory feedback areas tend to be topographically organized; central ganglia receive projections from various body parts and show general somatotopy (Walters et al., 2004; Vitzthum, Muller, & Homberg, 2002; Wong, Wang, & Axel, 2002). Interneurons become more common and constitute a key element in processing and integrating information.

The most advanced invertebrate class is the cephalopods—highly derived molluscs. They feature, on one hand, the highest centralization of the nervous system. On the other hand, next to the central nervous system (CNS) composed of the brain and two optic lobes, there is a large peripheral nervous system (PNS) of the body and the arms. The brain consists of 30–40 interconnected lobes with a high degree of cross-talk; yet, the interconnections appear less elaborate than in vertebrate brains (Young, 1971). Despite the high level of centralization and in contrast to vertebrate and insect brains, there is no obvious somatotopic arrangement in either motor or sensory areas.

The most prominent and most intelligent, and with the largest nervous system among cephalopods, is the octopus. The octopus has a unique embodiment—a flexible body and eight arms with virtually infinite degrees of freedom. Brain stimulation reveals that motor control is hierarchically organized into three functional levels. In the higher motor centres located in the basal lobes, microstimulation evokes complex movements which are, however, not somatotopically represented, but controlled

[1] The beginning of this section draws heavily on Zullo & Hochner (2011).

by parallel overlapping circuits representing individual motor programmes. The basal lobes receive inputs from the optic lobes and other sensory centres.

Yekutieli et al. (2005a) and Yekutieli, Sagiv-Zohar, Hochner, & Flash (2005b) developed a dynamic model of the octopus arm and used it to hypothesize the mechanism of how a reaching movement is generated. Despite the complexity of the arm, they found that in their model, only two control parameters suffice to fully specify the extension movement of the arm: (1) the amplitude of the activation signal; and (2) the activation travelling time. This hypothesis seems in line with electromyogram (EMG) recordings and other evidence from the real octopus. Larger amplitudes would result in the same kinematics, but larger forces, increasing the arm's stability against perturbations. For reaching directed at a particular target, two additional control parameters are necessary for the orientation of the arm base. Considering both the experimental and the simulation results, Yekutieli et al. (2005b) speculate that the octopus reaches toward a target using the following strategy:

(1) Initiating a bend in the arm so that the suckers point outward.
(2) Orienting the base of the arm in the direction of the target or just above it.
(3) Propagating the bend along the arm at the desired speed by a wave of muscle activation that equally activates all muscles along the arm.
(4) Terminating the reaching movement when the suckers touch the target by stopping the bend propagation and thus catching the target.

There are three kinematic control parameters (two angles for arm base orientation and one for movement speed) and one dynamic control parameter corresponding to muscle force.

However, the behavioural repertoire of the octopus is greater. Gutnick, Byrne, Hochner, and Kuba (2011) prepared a special setup where the octopus has to guide one of its arms through a maze to reach food in one of three branches marked by a visual cue. The octopus is capable of such 'hand–eye coordination'. Also, instead of the stereotypical largely feed-forward bend propagation, it uses a much slower—but possibly feedback-controlled—'search movement'. Finally, to bring food to the mouth, the arm is bent in a specific way, creating 'virtual joints' along it.

### 10.2.2 The body in the primate brain

It is not in my capacity or my goal to review the structure and function of vertebrate nervous systems. Instead, I will briefly discuss how the body is represented in the brain of primates, which include monkeys, apes, and humans. Again, reaching will serve as an example in which a body model of sorts needs to be employed.

The presence of various 'body maps' in the primate brain has fascinated scientists and the general public alike, spurred by the account of Head and Holmes (1911) and the discovery of the somatotopic representations (the 'homunculi') in the primary motor and somatosensory cortices (Leyton & Sherrington, 1977; Penfield & Boldrey, 1937). Neurological conditions and accounts of a whole range of illusions regarding own body perception (e.g., rubber hand illusion, out-of-body experience, apparition) generated both seminal research articles (e.g., Botvinick & Cohen, 1998; Lenggenhager, Tadi, Metzinger, & Blanke, 2007) and public interest. The attention devoted to the representations of the body in the brain has also led to numerous attempts at describing or defining them, and proposals of a variety of concepts, including superficial and postural schema (Head & Holmes, 1911), body schema, body image, corporeal schema, etc., have been put forth. One characteristic common to all these representations is their multimodal nature—they dynamically integrate information from different sensory modalities (visual, tactile, proprioceptive, vestibular, auditory) (Azañón et al., 2016), while not excluding motor information. However, the concepts of body schema, body image, and many others are umbrella notions for a range of observed phenomena, rather than a result of identification of specific mechanisms. The field is thus in a somewhat 'chaotic state of affairs' (Berlucchi & Aglioti, 2010), with limited convergence to a common view (Graziano & Botvinick, 2002; Holmes & Spence, 2004).

Reaching behaviour in primates bears some similarity to that in the octopus. A reaching movement has some high-level characteristics like the direction of a hand's movement in space, the extent of the movement (amplitude), the overall duration (movement time), and other parameters such as anticipated level of resistance to the movement (Schöner, Tekülve, & Zibner, 2018). Also, movement generation involves cooperation between the CNS and PNS. The exact mechanisms of motor control in humans and other primates are still debated (see, for example, Lisman, 2015). One view—the equilibrium point theory—posits that high-level descending motor commands modulate the peripheral reflex loops (such as the stretch reflex) and set a desired muscle length (Feldman, 2011). Compared to invertebrates, motor control in vertebrates, specifically mammals and in particular primates, becomes more 'cortical' and the motor cortex has the possibility of more direct control over the details of a particular movement, which is likely correlated with the need for dexterous manipulation.

Also, it may not be possible to decouple the movement preparation phase from its execution. The actual movement may be a product of couplings of feed-forward and feedback control that are necessary to understand the effects such as the uncontrolled manifold (Scholz & Schöner, 1999; Martin, Reimann, & Schöner, 2019).

It is also not completely clear whether a prerequisite for a reaching movement is localizing the effector—say the arm/hand—in space first. Through cortical microstimulation, Graziano was able to elicit stereotypical movements in the monkey, irrespective of the starting position (2006). However, Schöner et al. (2018) speculate that the stimulation could drive an update of the hand position first, which would be subsequently used for the movement generation—in line with Scott and Kalaska (1997) who found that neural tuning curves in the motor cortex depend on the arm's kinematic configuration.

In this chapter, I will specifically focus on a task in which localizing the own body cannot be circumvented—reaching to own body parts. For example, Lisman (2015) notes: 'More difficult to imagine is how one can reach one's ear lobe without visual guidance. This requires knowledge of the position of the target and the position of the limbs. It has only recently become clear that there is indeed a population code that represents hand position (Hauschild et al., 2012), but surprisingly little is known about the muscle and joint signals that allow this computation to be made (Weber et al., 2011).'

### 10.2.3 Robot body models

The world of robots and their body models is completely different. The first striking difference is that describing it is significantly easier; robot body representations and control schemes are designed by engineers and are thus very transparent and, unlike in the brain, we have complete access to all information. Let us take the iCub humanoid robot as an example. The iCub (Metta et al., 2010) (see Figure 10.1) is a baby humanoid robot that was designed after a 4-year-old child—with similar body proportions, kinematic structure, and sensorimotor capacities. At the same time, it is, to a large extent, a product of (great) engineering.

In Figure 10.1A, there is a cartoon of the robot, side by side with its computer-aided design (CAD) model (see Figure 10.1B) and the basic kinematic structure (joints and links) (see Figure 10.1C). Complete knowledge of the robot structure can be used to obtain a mathematical description of the robot's kinematics. This essentially corresponds to a sequence of coordinate transformations between all the reference frames in Figure 10.1D. Every such transformation consists of three translations and three rotations and

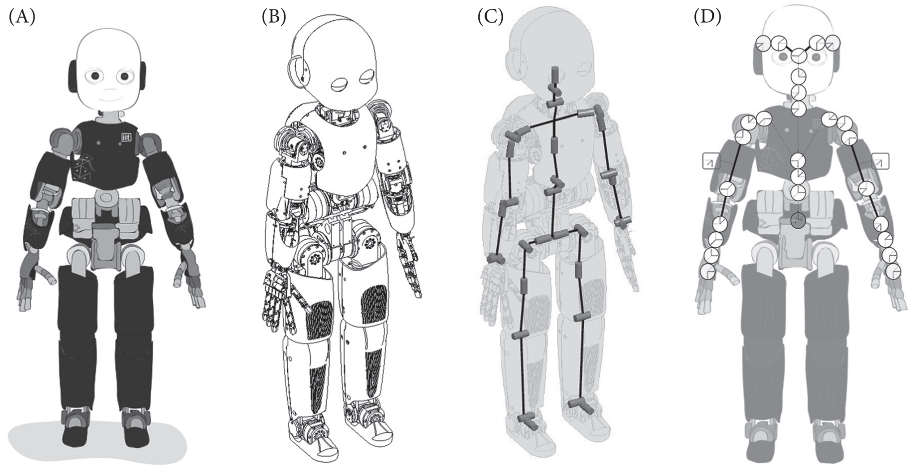


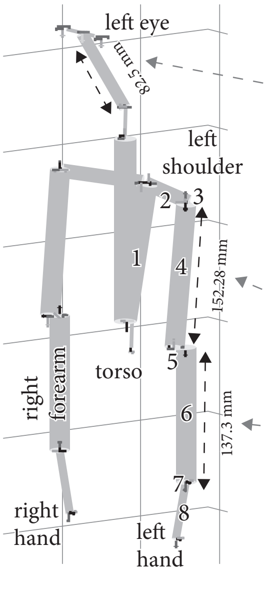
**Figure 10.1** The iCub humanoid robot. (A) Cartoon of the robot. (B) CAD model. (C) Kinematic structure. (D) Reference frames in upper body.

*Image credits*: (A) iCub cartoon: Reproduced courtesy of Laura Taverna, Italian Institute of Technology. (B) (C) iCub kinematic structure: Reproduced with permission from (Parmiggiani, et al., 2012), and courtesy of Alberto Parmiggiani. (D) Reproduced courtesy of Jorhabib Eljaik.

has thus six degrees of freedom (DoFs). However, as the robot structure is subject to specific constraints in three-dimensional space, four parameters suffice to characterize the transformation between consecutive links/joints. This is the essence of the Denavit-Hartenberg convention in which every link *i* (imagine getting from the elbow to the wrist along the forearm, for example) is described by four parameters: two lengths $a_i$ and $d_i$, and two angles $\alpha_i$ and $o_i$. In the iCub—and in most robots for that matter—all joints are revolute and have a single rotation axis, i.e., a single rotational DoF, like the human elbow (a 'hinge joint'). Figure 10.2 shows a schematic of this representation for the upper body of the robot.

### 10.2.3.1 Forward kinematics and inverse kinematics

The robot model describes the fixed characteristics of the robot body (long-term or 'offline' body representation; see later). The model in Figure 10.2 can be transformed into equations, whereby the coordinate transformation needed to get from one link to the next (e.g., from elbow to wrist) can be obtained as a simple matrix multiplication. To go over more links/joints (e.g., from torso to hand), these multiplications are simply sequenced. In the canonical form, this will only work for a single posture of the robot, like the one in Figure 10.1. To know where the body currently is in space, it has to be combined with the robot 'proprioception'—the joint angle values. These are plugged into the equations as one of the orientation parameters ($o_i$). This operation—combining current joint angle values with the robot model—is known as *forward kinematics*, which provides a mapping from *joint space* to *Cartesian space* (also called



| Link(i) | a(i) [mm] | d(i) [mm] | α [rad] | *o* [rad] |
|---|---|---|---|---|
| 1 | 2.31 | −193.3 | −π/2 | π/4 |
| 2 | 33 | 0 | π/2 | π/4 |
| 3 | 0 | 1 | −π/2 | π/4 |
| 4 | −54 | 82.5 | −π/2 | π/4 |
| 5 | 0 | −34 | −π/2 | 0 |
| 6 | 0 | 0 | π/2 | −π/4 |

TABLE I

DH Parameters – Head and Left Eye

| Link(i) | a(i) [mm] | d(i) [mm] | α [rad] | *o* [rad] |
|---|---|---|---|---|
| 1 | 23.36 | 143.3 | π/2 | 105 * π/180 |
| 2 | 0 | 107.74 | −π/2 | π/2 |
| 3 | 0 | 0 | π/2 | −π/2 |
| 4 | 15 | 152.28 | −π/2 | 75 * π/180 |
| 5 | −15 | 0 | π/2 | 0 |
| 6 | 0 | 137.3 | π/2 | −π/2 |
| 7 | 0 | 0 | π/2 | π/2 |
| 8 | 62.5 | −16 | 0 | 0 |

TABLE II

DH Parameters – Left Arm

**Figure 10.2** iCub kinematic model—upper body. (Left) Matlab visualization of the kinematic chains and reference frames. (Right) Denavit-Hartenberg (DH) parameters for the head and left eye and left arm. Correspondences for certain link lengths are marked.

*operational space* or *task space*). Thus, transformations between, say, hand frame, body frame, or eye frame can be readily obtained.

For a robot to reach to a specific position and with a specific orientation in Cartesian space—no matter in which reference frame the target is expressed as it can be transformed to the base frame—an inverse mapping is needed, from Cartesian space to joint space, i.e., to acquire the joint angles of the robot arm when the end effector (the hand) contacts the target. This is dealt with by *inverse kinematics*. Unlike forward kinematics, this is a harder problem and not just a matter of substituting for current joint angle values into the equations. Reaching for a target in three dimensions with a specific orientation constraints the robot pose in six dimensions (three positions and three orientations). Hence, a minimum of six DoFs—six joints—is required on the robot part. For six DoF manipulators—robot arms with six rotational joints—that have a specific geometric structure, a closed-form solution can be obtained. That is, a solution can be found instantaneously. However, in general, one has to resort to numerical methods. Robots with more than six DoFs have multiple ways of reaching for the target and hence are called *redundant manipulators*. Additional criteria are needed to choose among the solutions. In the iCub, there are seven DoFs in every arm and three additional ones in the torso. The manipulator is thus highly redundant. Inverse kinematics is solved numerically by using a non-linear optimizer (Pattacini, Nori, Natale, Metta, & Sandini, 2010).

#### 10.2.3.2 Motion control

Inverse kinematics provides the joint space configuration for the robot in the final position at the target. However, it does not automatically deal with the trajectory—joint and end effector positions in time—needed to move between the initial and final positions. *Trajectory generation* constitutes its own discipline that deals with planning such smooth movements using different interpolation methods, for example. Once the trajectory in joint space is planned, low-leveI motor controllers in every joint can be used to bring about the desired movements in time. In the iCub, Pattacini et al. (2010) designed a bio-inspired dynamical systems-based controller that produces smooth, minimum jerk trajectories in which the end effector (the hand) follows a quasi-straight line.

#### 10.2.3.3 Dynamics

While kinematics deals with positions, velocities, and accelerations, dynamics deals with equations of motion and forces that are needed to produce a particular movement. The reader is referred to any robotics textbook on the topic. For the iCub, Nori et al. (2015) provide an example.

## 10.3 Characteristics of body representations

In this section, I will attempt to compare the most important features of body representations in animals, humans, and robots. First, a note on terminology is in order. I use ‘body representations' as a general concept that should encompass both body schema and body image and possibly other body-related notions. However, by using the word 'representation', it is not my goal to take a stance in the philosophical debate on representationalist versus sensorimotor approaches to body awareness (de Vignemont, 2015). The account will admittedly be biased toward the 'representationalist' viewpoint (e.g., Carruthers, 2008; Longo et al., 2010)—also because I come from robotics and computer science—and will not address the phenomenological perspective and the 'lived body' (Merleau-Ponty, 1945). At the same time, I also fully endorse the 'sensorimotor approach' and I want to avoid, or at least reflect upon, imposing the representationalist stance typical of robotics and (good old-fashioned) artificial intelligence (Haugeland, 1985) onto the biological 'body in the brain' (see also our attempt in Hoffmann et al. (2017)). Webb (2006) provides a useful clarification of the terms *transformation*, *encoding*, and *representation*. The word r*epresentation* should be reserved to the strong notion of standing in for something—properties or states of the body that can be manipulated also when the body itself cannot be used or sensed directly. Sometimes, the body can be used directly—imagine the reaching in the octopus discussed above—and it is probably more natural to think that the 'brain is in the body' and does not have to have it all represented, rather than the 'body in the brain' is embodied (cf. a discussion in Alsmith & de Vignemont (2012)).

According to de Vignemont (2015), there are three principal *taxonomies of body representations*. I will list them below, together with their relationship to the account in this book section:

(1) The triadic taxonomy based on the 'format' of body representations (Schwoebel & Coslett, 2005) distinguishes:
   (a) Sensorimotor body representation (also known as body schema).
   (b) Visuo-spatial body representation (or body structural description).
   (c) Conceptual body representation (or body semantics).

   My account will span roughly the first two, leaving the conceptual body representation aside.

(2) The functional dyadic taxonomy (Dijkerman & De Haan, 2007; Gallagher, 2006; Paillard, 1999), based on the perception–action model of vision ('ventral stream' or 'what' versus 'dorsal stream' or 'where/how') (Milner & Goodale, 2006), distinguishes:
   (a) Body schema—sensorimotor representations of the body used for action planning and control.
   (b) Body image—lacking a unifying positive definition (Berlucchi & Aglioti, 2010; de Vignemont, 2010); comprises all the 'other' (than body schema) representations about the body that are not used for action: perceptual, conceptual, or emotional.

   My account will focus on the representations for action. Robot body models are also primarily geared toward action. However, interestingly, as their models are engineered from the outside, they do carry a lot of the flavour that is typical rather of the 'what' pathways that care about semantics of objects, etc.

(3) The temporal dyadic taxonomy (Carruthers, 2008) is based on the dynamics of body representations and contrasts:
   (a) Long-term or 'offline' body representations, such as limb size—what the body is usually like. These are relatively stable in adulthood and may include some innate components about body structure (e.g., two arms and two legs). Carruthers (2008) also argues 'that the offline body representation must be an integrated representation, a failure to integrate leads to body integrity identity disorder.' 'That is, it must represent the body as a single thing, rather than a collection of parts.'
   (b) Short-term or 'online' representations of the body as it is currently, such as in which posture, constantly updated on the basis of afferent and efferent information.

   In this case, both are obviously equally relevant. From the robotics perspective, the *long-term body representation* can be equated with *body model*. The short-term representation would be the *body state*. I will focus more on the long-term body representations.

To give the discussion concrete contours, let us look at the example in Figure 10.3. Figure 10.3A depicts a scenario from a series of studies on infants where they were observed reaching for a vibrotactile stimulus (buzzer) (Hoffmann et al., 2017; Leed, Chinn, & Lockman, 2019). Two components seem necessary: (1) localizing the stimulus on the body; and (2) reaching for it. The centre in Figure 10.3A depicts a possible decomposition into blocks. The left part (up to the 'localization' blocks) (Tamè, Azañón, & Longo, 2019) deals with the 'localization' or somatoperceptual processing part. Once the target is localized (*spatial localization of touch* block), it can be adopted as the *reaching target* (illustrated with a '~'), and the motor action can be prepared and executed. The different shapes for the blocks were reproduced after (Longo et al., 2010) as follows: 'inputs are depicted as diamond shapes, body representations as ovals, and perceptual processes as rectangles'. It is interesting to look at this from the temporal taxonomy perspective; it seems that only the long-term or offline modules count as representations here. The percepts, however, represent 'states of the body' and can be viewed as short-term or online body representations. Heed, Buchholz, Engel, and Röder (2015) offer a different conceptualization of this scenario, also known as *tactile remapping*—the transformation of a coordinate in a skin-based reference frame into a coordinate in an external reference frame by integration of posture information. Heed et al. actually present three variants of the schematics (not reproduced here): (a) remapping view; (b) integration view; and (c) sensorimotor contingency view. In (b) and (c), the arrows between the 'localization' part and the 'action' part point in both directions, which illustrates another important aspect of the problem—a sequential processing view and decoupling of the perception and action parts may not be justified, something we touched upon in Section 10.2.2. Analysis of the infant reaching behaviour in the buzzer experiments shows that looking at the target and reaching often happen simultaneously; in addition, for targets on the arms/hands, both the limb with the target and

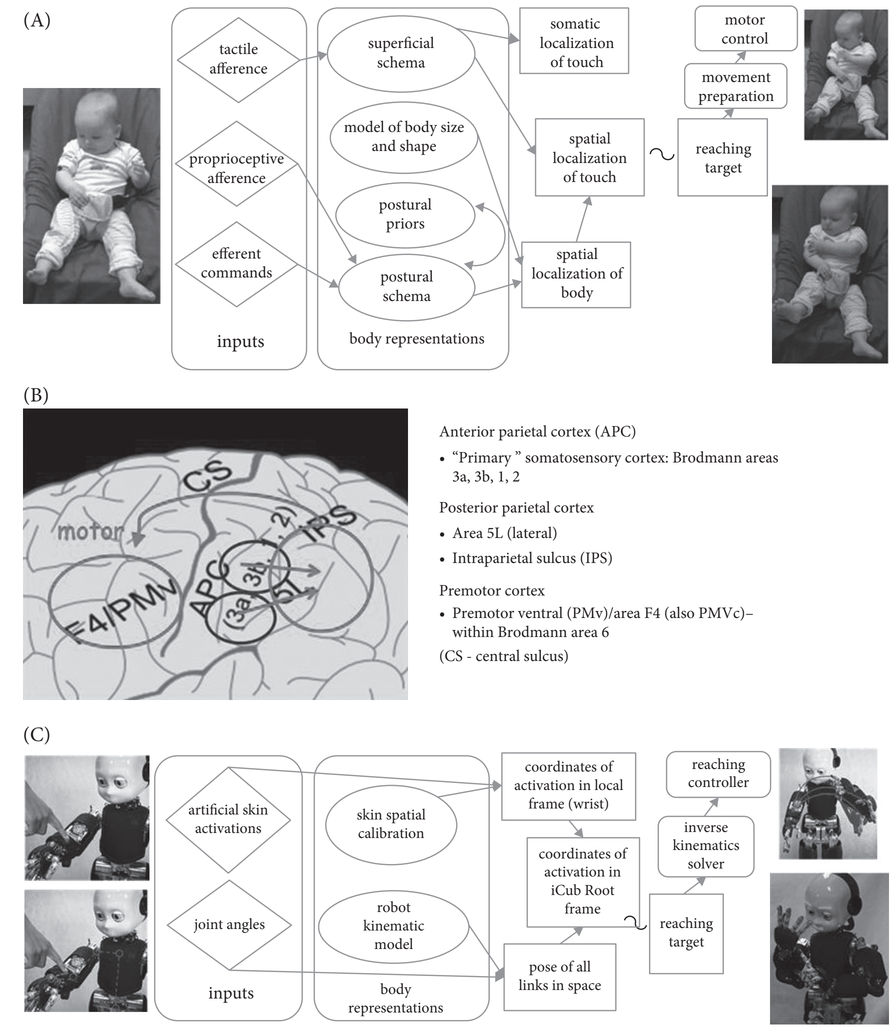

(A)
tactile afference
proprioceptive afference
efferent commands
inputs
superficial schema
model of body size and shape
postural priors
postural schema
body representations
somatic localization of touch
spatial localization of touch
spatial localization of body
~
reaching target
movement preparation
motor control
(B)
CS
motor
F4/PMv
APC
(3a
3b
, 2)
5L
IPS
Anterior parietal cortex (APC)
• "Primary " somatosensory cortex: Brodmann areas 3a, 3b, 1, 2
Posterior parietal cortex
• Area 5L (lateral)
• Intraparietal sulcus (IPS)
Premotor cortex
• Premotor ventral (PMv)/area F4 (also PMVc)– within Brodmann area 6
(CS - central sulcus)
(C)
artificial skin activations
joint angles
inputs
skin spatial calibration
robot kinematic model
body representations
coordinates of activation in local frame (wrist)
coordinates of activation in iCub Root frame
pose of all links in space
~
reaching target
inverse kinematics solver
reaching controller

**Figure 10.3** Reaching to the body. (A) Scenario with reaching for vibrotactile stimuli in infants (Hoffmann et al., 2017). Photos show an infant presented with the stimulus (left), in the process of reaching (top right), and in the final posture (bottom right). Centre: a conceptual model of somatosensory processing (left part, up to '~', adapted from Longo, Azañón, and Haggard (2010) and Tamè, Azañón, & Longo, (2019)) and reaching action. (B) Schematic illustration of cortical areas that may be responsible for bringing about the behaviour. (C) Similar scenario on the iCub humanoid robot: tactile stimulus (left), motor action (top right), and final configuration (bottom right). Centre: block diagram illustrating the modules used to generate the 'self-touch' behaviour (Roncone, Hoffmann, Pattacini, & Metta, 2014).

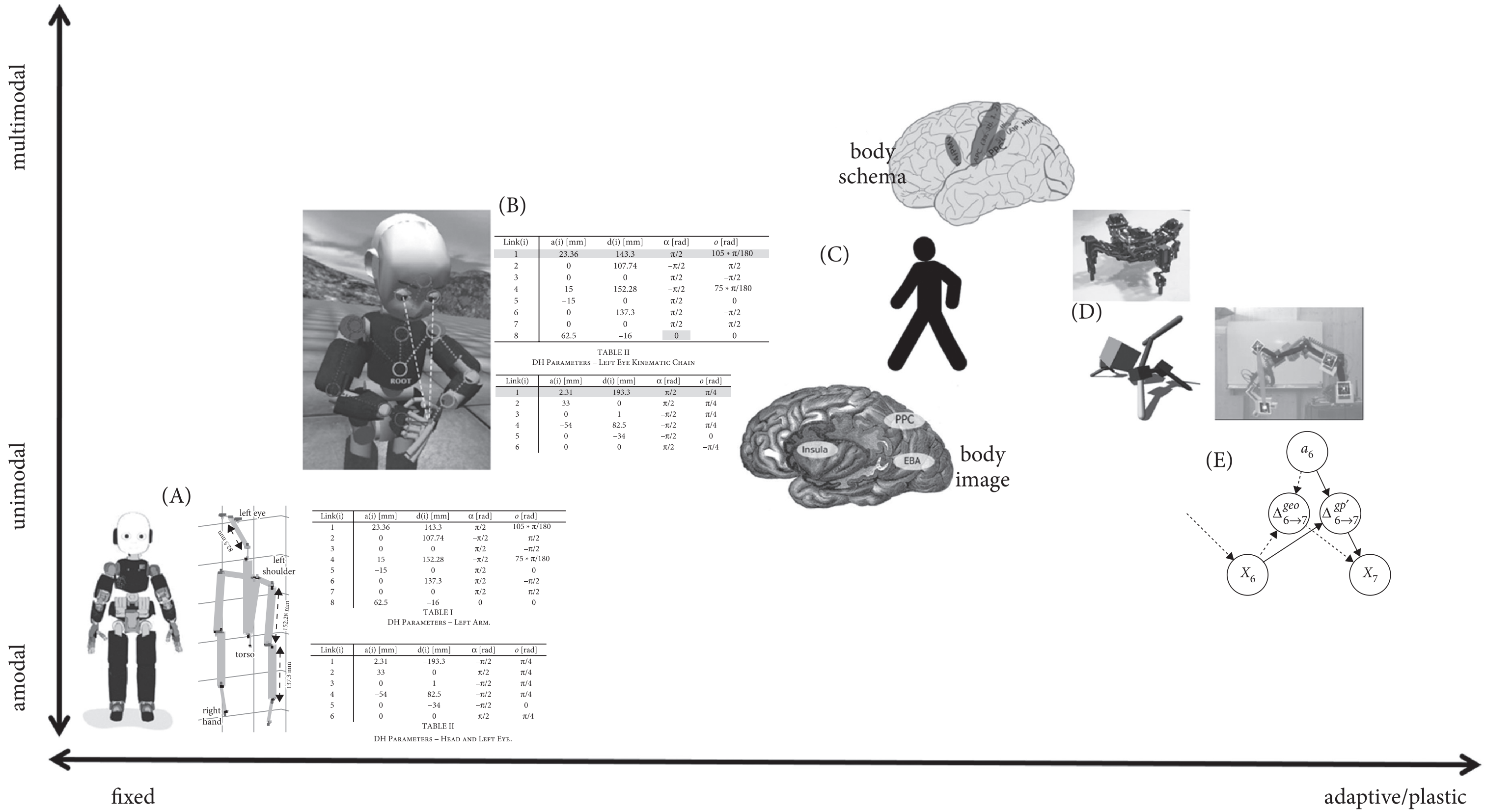
*Image credits*: (A) Flowchart adapted from (Tamè, Azañón, & Longo, 2019) under a Creative Commons Attribution License (CC BY) (https://creativecommons.org/licenses/by/4.0/) and with permission from Longo, M. R., Azañón, E., and Haggard, P. (2010). With additional blocks illustrating the motor part. (B) Brain image adapted from Hugh Guiney/Creative Commons Attribution-Share Alike 3.0 Unported license (CC BY- SA 3.0) (labels added). (C) Reproduced courtesy of Alessandro Roncone and Matej Hoffmann.

the reaching contralateral arm often move simultaneously (Chinn, Hoffmann, Leed, & Lockman, 2019). The more general point of recurrent connections also relates to the state estimation problem; the 'percepts' may not be the result of a single pass that combines information from different sources but may be the result of conflict reconciliation where activations need to flow back and forth.

Figure 10.3B schematically illustrates some of the cortical areas of the monkey brain that may be involved in generating this behaviour. Finally, Figure 10.3C shows an instantiation of a similar scenario on the iCub humanoid robot. There is one important difference—this is not a conceptualization of the behaviour; this is an actual pipeline that has been used in Roncone, Hoffmann, Pattacini, and Metta (2014) (for the video, see https://youtu.be/pfse424t5mQ). The blocks correspond to actual pieces of software. Surprisingly, there is quite a good match with the schematics in Figure 10.3A, which may be because the schematics of Longo and colleagues is somewhat classical and thus compatible with engineering models that—mostly for practical reasons—typically follow a modular design and 'sense–think–act' logic.

There are almost countless characteristics of body representations that we can think of. In what follows, I will sketch some of the important ones, focusing in particular on those where contrasting the biological and robot worlds can bring the most insight. Hence, I will, for example, leave the unconscious versus the conscious aspect aside—as discussed above, the focus here is on the sensorimotor level. I will take a number of examples from biology and robotics, developing the ideas in Hoffmann et al. (2010) and Hoffmann & Pfeifer (2018). I will also, sometimes quite speculatively, attempt to chart the body schema and body image onto the hypothetical axes. Body image will largely stand for visuo-spatial representation of the body or body structural description—body percept rather than body concept. The brain areas involved are also only schematically illustrated.

In Figure 10.4, the iCub humanoid robot and the kinematic model of its upper body is depicted in panel (A). The model has been essentially handcrafted by following the Denavit-Hartenberg convention and supplying the corresponding lengths and angles from the CAD model of the robot. Panel (B) adds the possibility to calibrate the same model automatically as the robot exploits self-observation and self-touch configurations (Stepanova, Pajdla, & Hoffmann, 2019). Panels (D) and (E) depict other examples of robot self-calibration. Bongard, Zykov, and Lipson (2006) used a quadrupedal machine continuously 'self-modelling' itself. The robot model had a special nature; it consisted of a physics-based simulator with a copy of the robot's limbs, motors, sensors, and even the environment. Within this engineered 'world and body model', the robot would search for its kinematic structure by comparing the actions and their sensory consequences from the physical world with those in the simulator. Sturm, Plagemann, and Burgard (2009) had a robot arm observe 'itself' using a camera and infer its model—learning the structure of a Bayesian network—from motor actions and observations in the camera. Panel (C) schematically illustrates humans and the body schema and body image.



| Link(i) | a(i) [mm] | d(i) [mm] | α [rad] | *o* [rad] |
|---|---|---|---|---|
| 1 | 23.36 | 143.3 | π/2 | 105 ∗ π/180 |
| 2 | 0 | 107.74 | −π/2 | π/2 |
| 3 | 0 | 0 | π/2 | −π/2 |
| 4 | 15 | 152.28 | −π/2 | 75 ∗ π/180 |
| 5 | −15 | 0 | π/2 | 0 |
| 6 | 0 | 137.3 | π/2 | −π/2 |
| 7 | 0 | 0 | π/2 | π/2 |
| 8 | 62.5 | −16 | 0 | 0 |

TABLE I
DH Parameters – Left Arm.

| Link(i) | a(i) [mm] | d(i) [mm] | α [rad] | *o* [rad] |
|---|---|---|---|---|
| 1 | 2.31 | −193.3 | −π/2 | π/4 |
| 2 | 33 | 0 | π/2 | π/4 |
| 3 | 0 | 1 | −π/2 | π/4 |
| 4 | −54 | 82.5 | −π/2 | π/4 |
| 5 | 0 | −34 | −π/2 | 0 |
| 6 | 0 | 0 | π/2 | −π/4 |

TABLE II
DH Parameters – Head and Left Eye.

| Link(i) | a(i) [mm] | d(i) [mm] | α [rad] | *o* [rad] |
|---|---|---|---|---|
| 1 | 23.36 | 143.3 | π/2 | 105 ∗ π/180 |
| 2 | 0 | 107.74 | −π/2 | π/2 |
| 3 | 0 | 0 | π/2 | −π/2 |
| 4 | 15 | 152.28 | −π/2 | 75 ∗ π/180 |
| 5 | −15 | 0 | π/2 | 0 |
| 6 | 0 | 137.3 | π/2 | −π/2 |
| 7 | 0 | 0 | π/2 | π/2 |
| 8 | 62.5 | −16 | 0 | 0 |

TABLE II
DH Parameters – Left Eye Kinematic Chain

| Link(i) | a(i) [mm] | d(i) [mm] | α [rad] | *o* [rad] |
|---|---|---|---|---|
| 1 | 2.31 | −193.3 | −π/2 | π/4 |
| 2 | 33 | 0 | π/2 | π/4 |
| 3 | 0 | 1 | −π/2 | π/4 |
| 4 | −54 | 82.5 | −π/2 | π/4 |
| 5 | 0 | −34 | −π/2 | 0 |
| 6 | 0 | 0 | π/2 | −π/4 |

**Figure 10.4** Body representation characteristics I: plasticity and number of modalities. (A) iCub humanoid robot and its kinematic model. (B) iCub kinematic self-calibration using self-observation and self-touch (Stepanova, Pajdla, & Hoffmann, 2019). (C) Human and schematic illustration of brain areas important for body representations: body schema (see Figure 10.3B for details) and body image (posterior parietal cortex, PPC; extrastriate body area, EBA; insula) (after Berlucchi & Aglioti, 2010). (D) Four-legged machine learning its body structure (Bongard, Zykov, & Lipson, 2006). (E) Robot manipulator learning body structure from self-observation (Sturm, Plagemann, & Burgard, 2009).

*Image credits*: Figure revised and expanded from (Hoffmann & Pfeifer, 2018). (A) iCub cartoon: Reproduced courtesy of Laura Taverna, Italian Institute of Technology. (B) Reproduced with permission from (Stepanova, Pajdla, & Hoffmann, 2019) (C) Walking human: Public domain (https://commons.wikimedia.org/wiki/File:BSicon_WALK.svg). Body schema brain adapted from Hugh Guiney/ Attribution- ShareAlike 3.0 Unported (CC BY- SA 3.0). Body image brain: Public domain (https://commons.wikimedia.org/wiki/File:Gray731.png). (D) Reproduced with permission from (Bongard, Zykov, & Lipson, 2006), and courtesy of Josh Bongard, University of Vermont. (E) Reproduced with permission from (Sturm, Plagemann, & Burgard, 2009), and courtesy of Jürgen Sturm.

### 10.3.1 Plasticity

Let us first consider plasticity or adaptivity of body models. Traditional robot models are fixed—an industrial manipulator is shipped with its model. It may not be directly available to the customer, but it will be embedded in the robot controller which needs the robot model(s) for operation. Even industrial robots may require occasional (re)calibration, which can be performed using different routines. Less accurate or more complex robots, such as humanoids, may need recalibration more frequently. The work of Stepanova et al. (2019) (see Figure 10.4B) is one of many examples in which the robot kinematic model parameters are calibrated. The approach is rather straightforward; redundant information about the positions of certain body parts—from self-touch or self-observation in this case—drives learning; with two hands physically touching and both cameras observing the point, any mismatch between the position of the corresponding point—after remapping into a common frame of reference—generates an error term used to update the model (all but the grey parameters are calibrated). Body representations in primate brains (Figure 10.4C) are known for their plasticity on several timescales. First, body models need to be discovered by the brain, starting already in the fetal period (see the work of Kuniyoshi and colleagues on embodied computational models of this in Section 10.4). Second, body models need to adapt, as the body grows, for example. Third, they adapt or optimize when some body parts are frequently used in a specific task—when playing a musical instrument, for example. Fourth, they adapt also on very short timescales like when adapting to tools or when the subject is tricked by some of the numerous illusions, like the Pinocchio illusion. In the last case, it is rather a case of 'short-term body schema' adaptation—a state estimation process—rather than adaptation of the body model itself, although this may be happening simultaneously and some after-effects observed. Next to the temporal taxonomy (online versus offline body representations), I will speculate about the distinction between body schema and body image on the plasticity axis. It seems that most of the rapid recalibrations pertain to the body schema which draws more directly on the inputs from different modalities and their integration. Taking the rubber hand illusion (Botvinick & Cohen, 1998) as an example, the body schema adaptation is manifested in the proprioceptive drift; however, the participants also start to quickly incorporate the rubber hand into their bodies—'own' the rubber hand—which is typically associated with body image (and the insula). The suggested primacy of body schema over body image (Pitron, Alsmith, & de Vignemont, 2018) may also be relevant here—body schema adaptation may, in a second step, propagate to body image.

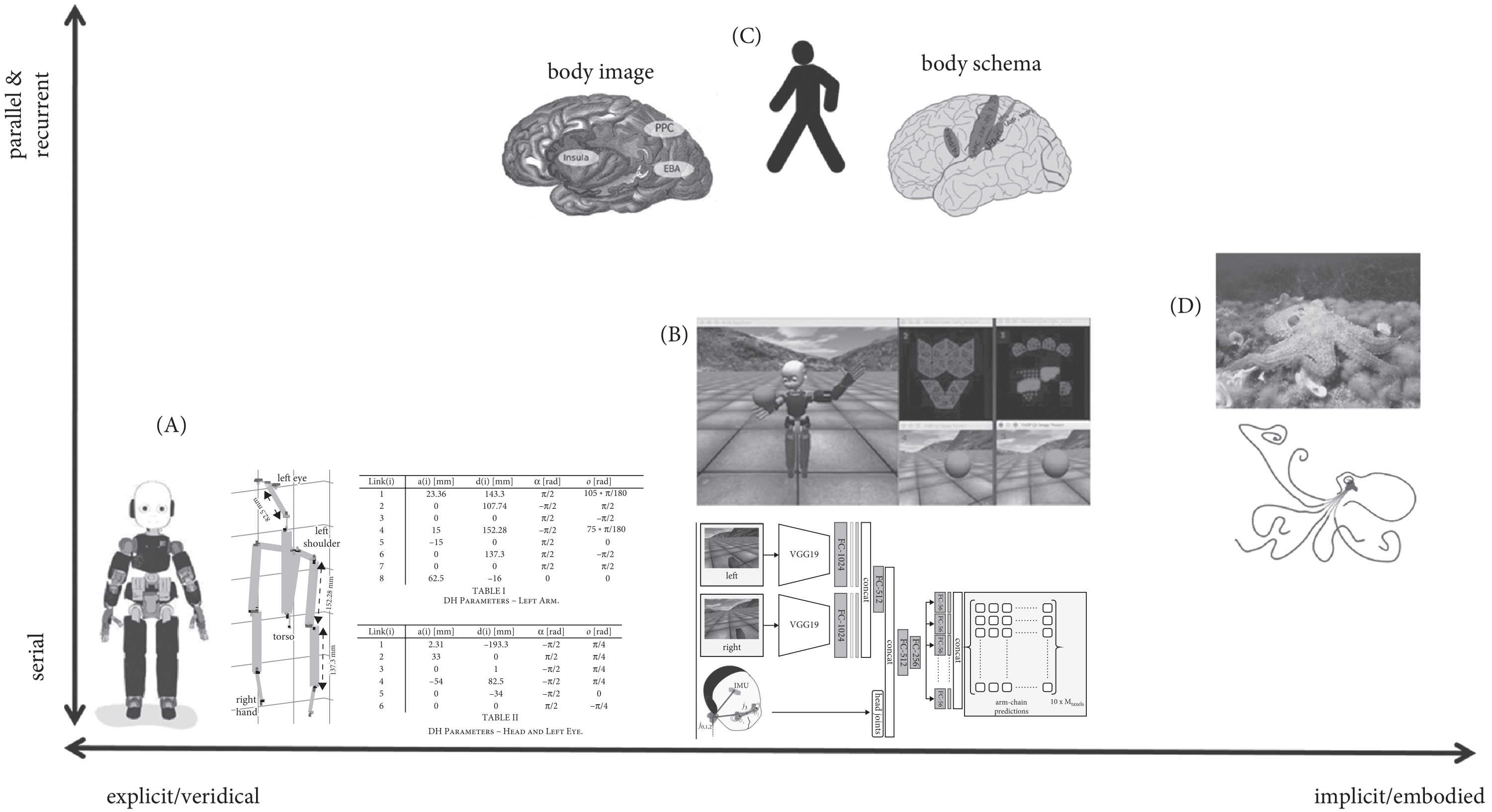
It seems that the brain is rather 'liberal' about the constraints imposed on the models and can be led into believing highly improbable things, like the 'nose elongation' during Pinocchio illusion. Robot models, on the other hand, tend to have quite strict constraints or bounds on the model parameters—capitalizing on the knowledge available from the outside—and would thus not fall for the illusions easily. At the same time, there are limits to the plasticity. Important evidence suggestive of innate and fixed—immune to experience—components of body models comes from the phantom limb phenomenon, which may be experienced following amputation, but also even in some subjects who congenitally lack limbs (e.g., Ramachandran & Blakeslee, 1998). That is, the basic body layout may be, to some extent, hard-wired in the model and immutable. The self-calibrating robots in panels (D) and (E) in Figure 10.4 move in that sense beyond this, as they are able to learn any topology of their body layout—that is why they are positioned more to the right on the plasticity axis. However, this is clearly an oversimplification—these robots surpass the brain plasticity in this single aspect only.

### 10.3.2 Multimodal nature of body models

Standard robot models are *amodal*—they do not depend on any sensory modality; they directly describe physical reality like the geometry of the body (see Figure 10.2 or Figure 10.4A). This holds in some sense also for all the other robot body models in Figure 10.4B, D, E. In Figure 10.4B, the robot model itself is identical to that in Figure 10.4A. The sensory modalities—proprioception, touch, and vision in this case—are needed to collect the redundant information about the body's position in space and update the model. This layer is separated from the model of the robot geometry itself. In Figure 10.4D, E, the situation is quite similar. Similarly to Figure 10.4B, Figure 10.4E features self-observation. In Figure 10.4D (Bongard et al., 2006), three modalities—touch, tilt, and clearance—are used to compare their values from the real robot with those from alternative body layouts in the simulator. Based on this, the case studies are localized on the 'modality' axis. Body representations in the brain are famous for their 'multimodality'. Azañón et al. (2016) review the multisensory contribution to body representations: visual, somatosensory (tactile and proprioceptive), vestibular, auditory, and nociceptive. Hence, 'body in the brain' scores highest on the 'multimodality axis' in Figure 10.4. A divide-and-conquer approach is used throughout the article by Azañón et al.—modality by modality. This is also discussed by the authors—such an approach is useful experimentally, but implausible in reality as there would be a lot of 'cross-talk' between the modalities. Indeed, the body representations are assumed to be, in some sense, unified or coherent. In light of the works on robot self-calibration, this begs the question of whether the brain arrives, in some sense, to an amodal, modality-independent model of the, say, body in space, onto which different sensory modalities converge. This resembles the *emulation theory of representation* proposed by Grush (2004) who uses the Kalman filter[2] metaphor—the amodal, long-term body model is a central representation, which also includes its relationship with individual sensory modalities. The filter can then perform state estimation—current state of the body—by executing a 'sensory update'.[3] To what extent this would be the case for the brain

[2] Coming from signal processing, a Kalman filter is an efficient recursive filter—device or process that removes some unwanted components or features from a signal—that estimates the internal state of a linear dynamic system from a series of noisy measurements.

[3] Grush, taking human arm as an example, contrasts an internal model in the form of a look-up table storing previous input-output sequences with an *articulated model*—a model that includes some variables corresponding to their counterparts in the musculoskeletal system (e.g., elbow angle, arm angular inertia, tension on quadriceps). Some of these variables can be measured (e.g., by stretch receptors) and these sensors can also be simulated in the emulator.



| Link(i) | a(i) [mm] | d(i) [mm] | α [rad] | *o* [rad] |
|---|---|---|---|---|
| 1 | 23.36 | 143.3 | π/2 | 105 ⋆ π/180 |
| 2 | 0 | 107.74 | −π/2 | π/2 |
| 3 | 0 | 0 | π/2 | −π/2 |
| 4 | 15 | 152.28 | −π/2 | 75 ⋆ π/180 |
| 5 | −15 | 0 | π/2 | 0 |
| 6 | 0 | 137.3 | π/2 | −π/2 |
| 7 | 0 | 0 | π/2 | π/2 |
| 8 | 62.5 | −16 | 0 | 0 |

TABLE I
DH Parameters – Left Arm.

| Link(i) | a(i) [mm] | d(i) [mm] | α [rad] | *o* [rad] |
|---|---|---|---|---|
| 1 | 2.31 | −193.3 | −π/2 | π/4 |
| 2 | 33 | 0 | π/2 | π/4 |
| 3 | 0 | 1 | −π/2 | π/4 |
| 4 | −54 | 82.5 | −π/2 | π/4 |
| 5 | 0 | −34 | −π/2 | 0 |
| 6 | 0 | 0 | π/2 | −π/4 |

TABLE II
DH Parameters – Head and Left Eye.

**Figure 10.5** Body representation characteristics II: explicit versus implicit and serial versus parallel. (A) iCub humanoid robot and its kinematic model. (B) iCub learning to reach using deep learning. (C) Human and schematic illustration of brain areas important for body representations: body schema (see Figure 10.3B for details) and body image (posterior parietal cortex, PPC; extrastriate body area, EBA; insula) (after Berlucchi & Aglioti, 2010). (D) The octopus and schematic of its nervous system.

*Image credits*: (A) iCub cartoon: Reproduced courtesy of Laura Taverna, Italian Institute of Technology. (B) Reproduced with permission from (Nguyen, Hoffmann, Pattacini, & Metta, 2019). (C) Walking human: Public domain (https://commons.wikimedia.org/wiki/File:BSicon_WALK.svg). Body schema brain adapted from Hugh Guiney/ Attribution- ShareAlike 3.0 Unported (CC BY- SA 3.0). Body image brain: Public domain (https://commons.wikimedia.org/wiki/File:Gray731.png). (D) Photo: Common octopus- albert kok/CC BY-SA (https://creativecommons.org/licenses/by-sa/3.0). Octopus nervous system: Jean-Pierre Bellier/CC BY-SA (https://creativecommons.org/licenses/by-sa/4.0).

remains unclear. Different distortions of body perception—inheriting some properties of the imperfect representations of individual modalities like the somatosensory homunculus—seem to suggest that the brain has not synthesized a perfect amodal model of its body (see, for example, Fuentes, Longo, & Haggard, 2013 and Longo, 2015). Finally, we can probably say that the body schema is more strongly multimodal, while the body image—at least as a pictorial description of the body based on mainly visual exteroception—would be less multimodal.

### 10.3.3 Explicit/veridical versus implicit/embodied body models

Robot body models are explicit; it is clear what in the model corresponds to what in the body (e.g., a certain parameter to the length of the left forearm) (see Figure 10.2 and Figure 10.5A). They are also objective and veridical; the parameters should be the true physical values of the quantities (lengths, angles, masses, etc.). In the biological realm, representations in general are not like that. Of course, this depends on the school of cognitive science, but there seems to be growing consensus about the embodied and action-oriented nature of cognition (e.g., Engel, Maye, Kurthen, & König, 2013). This should hold for representations of the body as well (Alsmith & de Vignemont, 2012). 'What the nervous system needs to do, in general, is to transform the input into the right action' (Webb, 2006). We can take again the example of reaching behavior. As discussed in Section 10.2.1, the octopus is able to reach for visual targets, but it may not know—and may not need to know—how long its arm is or where it is exactly in space. Orienting the base of the arm and propagating the bend until contact is detected by the suckers may well suffice. This is *embodied action* and may be in line with the fact that no specific body representation—somatotopic or other—sites were found in its nervous system. The need to represent the body, its state, and the complex inverse kinematics and dynamics has been largely offloaded to embodiment—the properties of the muscular hydrostat, supported by the PNS and low-dimensional inputs from the CNS. This is also in line with the thinking of Cisek and Kalaska (2003) who highlight the importance of the online, dynamically generated character of movement generation. At the same time, they also point out that due to conduction delays inherent to the sensorimotor system, purely feedback control is limited, or at least slow, perhaps manifested in the octopus experiments of Gutnick et al. (2011). Successful action is also the only criterion for the 'quality' of what is represented about the animal's body in its brain; there is no need for any objective or veridical representation. Hence, the octopus has been positioned on the far right of the x-axis in Figure 10.5. Insects would be even farther right on this axis, with their 'body in the nervous system' so implicit that we may be reluctant to call this representation altogether.

Similar arguments hold for primate brains, but to a lesser extent. Numerous sites dedicated to representing the body were found (see Figure 10.3B and Figure 10.5C). For reaching, some evidence is suggesting that common reference frames encoded in neurons in the posterior parietal cortex may be used for movement plans (Cohen &

Andersen, 2002), for example. However, 'neurophysiological studies routinely fail to find a significant population of cells whose activity explicitly encodes the output of that transformation in a unique coordinate system. Instead the output may be implicitly embedded in the distributed pattern of activity across the population … ' (Cisek & Kalaska, 2003; see also Heed et al., 2015). This is perhaps even more prominent in the motor cortex where it is still being debated what it is the cells are specifying about reach—muscle force, movement direction, or a more abstract end goal of muscle action (Lisman, 2015). Somatotopy is rather functional than based on the spatial relationships of the body. The embodied perspective is also appropriate here (see Corbetta, Wiener, Thurman, & McMahon, 2018, for example). At the same time, compared to the octopus, much more of the body seems more explicitly represented. Interestingly, Longo (2015) also considers the implicit–explicit axis for body representations and draws a line roughly between the 'body schema' and the 'body image'. In tasks more related to action and where humans do not consciously represent their body, the body models seem more implicit and also less accurate. These representations may also be dominated by somatosensation and inherit some of the distortions typical of the 'somatosensory homunculi'. Conversely, tasks that relate to conscious perception of our body seem to draw on more explicit representations that are also more accurate/veridical (e.g., image of our hand). This is schematically illustrated in Figure 10.5C.

Finally, works in robotics can also move toward more implicit models—this is in line with the current advent of deep learning and end-to-end architectures. One example from the iCub robot is shown in Figure 10.5B (Nguyen, Hoffmann, Pattacini, & Metta, 2019). From motor babbling experience, the robot learns to associate the head and eye configuration and stereo-image input with arm joint configuration required to reach, together with the body part that will contact the object. The complete mapping is implicitly represented in a deep convolutional neural network. Only the feed-forward component is represented, not the motor execution, which is performed using traditional methods.

### 10.3.4 Serial versus parallel processing

Another axis that separates robot body models from their biological counterparts is serial/sequential versus parallel processing (y-axis in Figure 10.5). This has to do with the design, but also with the computing substrate. Robot control architectures tend to have a serial 'sense–think–act' design; in addition, their control systems run on substrates derived from the classical von Neumann architecture. Biological brains are known for their massively parallel processing. This is nicely illustrated in the example of visually guided behaviour in Figure 10.6 (Cisek & Kalaska, 2003). Interestingly, Pitron et al. (2018) suggest some serial aspect of body representations whereby the body schema would be feeding the body image.

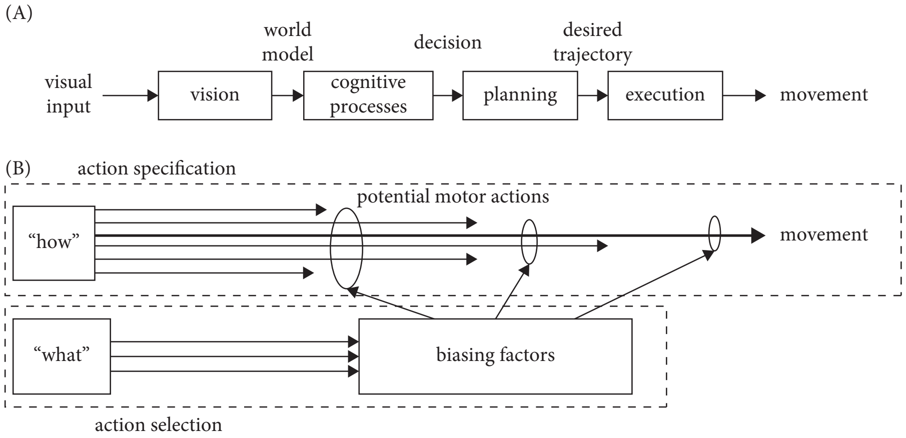


**Figure 10.6** Sequential versus parallel processing for visually guided behaviour. (A) The traditional 'sequential processing' model of visually guided behaviour. In this model, visual input is used to construct a model of the world which is used to make decisions. After decisions are made, a desired trajectory is generated and executed. (B) Schematic representation of the 'specification and selection' architecture for visually guided behaviour. Under this view, visual information has two different roles: specifying the parameters of potential motor actions; and defining criteria which bias competition among those potential actions until a single action is selected. These biasing factors include attention, behavioural relevance, prior reinforcement, required effort, behavioural context, learnt associations, motivations, long-term behavioural objectives, desired outcomes, and any other factor which influences action selection. The processes of specification and selection occur in parallel and continue even during overt movement. A striking feature of this architecture is the absence of a central model of the visual world.

Reproduced with permission from, Arbib, Michael A., ed., The Handbook of Brain Theory and Neural Networks, second edition, Figure 1 from 'Reaching Movements: Implications for Computational Models', © 2002 Massachusetts Institute of Technology, by permission of The MIT Press.

A second aspect is the presence of recurrent connections. In robot control systems, the flow of information is sequential and also unidirectional. Conversely, in neural systems, recurrent connections are ubiquitous. In primate brains, due to their complexity, hierarchies are formed and information flows back and forth, combining the bottom-up and top-down influences at different stages of the hierarchy. This is valid universally, with the circuitry responsible for representing the body not being an exception. The recurrent nature is also more pronounced in primate nervous systems than in cephalopods, say—mainly due to the overall difference in the number of neurons and layers of processing. This difference has also implications on the nature of the online, or short-term, body representations. In traditional robot frameworks, state estimation will be at the top of the sensory processing part, perhaps combined with priors from the model. However, in the brain, state estimation is a highly dynamic, continuous process combining multisensory integration, top-down priors, etc.—through recurrent loops.

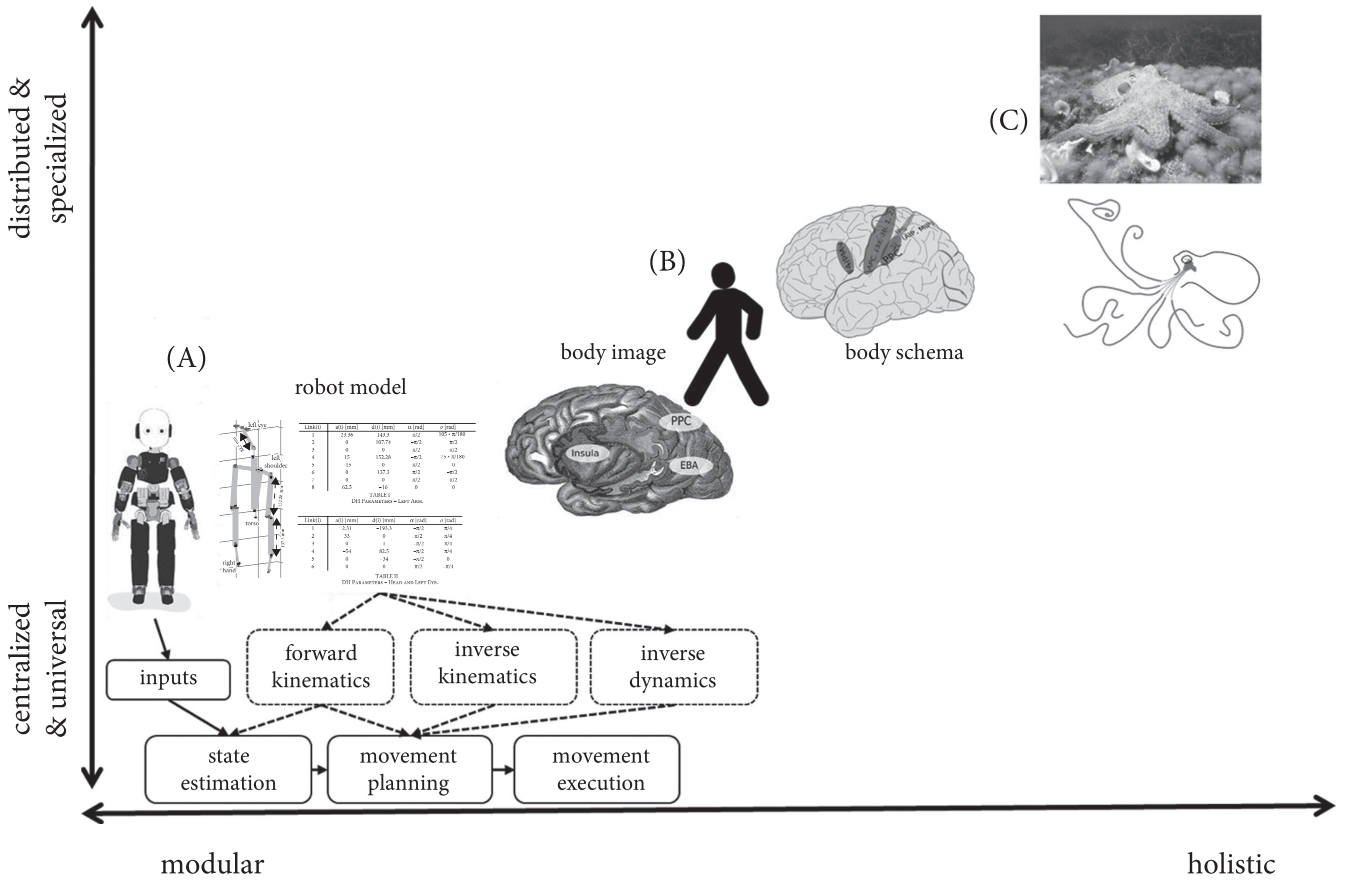
Finally, the robot case study (see Figure 10.5B) features a purely feed-forward, hence sequential, neural network.

### 10.3.5 Modular versus holistic and centralized versus distributed representations

A final graphical attempt to contrast body models in robots and animals is depicted in Figure 10.7. At first glance, it would seem intuitive that robot models are centralized, while 'body in the brain' is highly distributed. There is typically only one body model for a robot. In contrast, in the brain, there are numerous distributed, incomplete body representations. In this sense, there is some overlap between centralized versus distributed and explicit versus implicit distinction. However, centralized in this case does not imply monolithic. In fact, robot body models and associated control modules are highly modular, as shown in Figure 10.7A. There are distinct modules like forward/inverse kinematics and dynamics that may draw from the same robot model and be recruited for different purposes like state estimation, movement planning, etc. There would be typically only one module of every kind (imagine a software library) providing this functionality upon request. The representations/modules will thus be universal (as opposed to task-specific) and not overlapping. In nervous systems, on the other hand, there would be often complete sensorimotor loops specialized in a particular task (and hence task-specific rather than universal). In that sense, every such representation would be, in some sense, holistic. Their functionality may also be partially overlapping or redundant. Population coding is a good example; it is striking how unspecific the receptive fields of individual neurons often are (e.g., Georgopoulos, Kalaska, Caminiti, & Massey, 1982). Also, many tasks may require some form of representation of the hand in space, say, but it cannot be excluded that this will be implicitly 'implemented' multiple times. This distributed, specialized, and holistic nature is highest for the nervous system of the octopus (see Figure 10.7C); regarding body models in primates (see Figure 10.7B), the body schema as evolutionarily older and more 'low-level' is on these axes expected to be closer to the octopus, whereas the body image somewhat closer to the engineered body models.

## 10.4 Robots as embodied models of body representations

How can the biological disciplines like cognitive psychology and neuroscience profit from the viewpoint of robot models? The engineering perspective provides a mature analytical machinery that deals with the relevant problems and this can be certainly exploited. For example, consider again the 'multimodal nature of body models' discussed in Section 10.3.2. Next to the Kalman filter metaphor employed by Grush (2004) to describe how estimating the current body state might work, mathematical methods can be also employed to answer fundamental questions like under what exact



igure 10.7 Body representation characteristics III: modular versus holistic and centralized versus distributed. (A) iCub humanoid robot, its kinematic model, and ›me control modules. (B) Human and schematic illustration of brain areas important for body representations: body schema (see Figure 10.3 for details) and body ıage (posterior parietal cortex, PPC; extrastriate body area, EBA; insula) (after Berlucchi & Aglioti, 2010). (C) The octopus and its nervous system.

*ıage credits*: (A) iCub cartoon: Reproduced courtesy of Laura Taverna, Italian Institute of Technology. (B) Walking human: Public domain (https://commons.wikimedia.org/wiki/File:BSicon_WALK. g). Body schema brain adapted from Hugh Guiney/ Attribution- ShareAlike 3.0 Unported (CC BY- SA 3.0). Body image brain: Public domain (https://commons.wikimedia.org/wiki/File:Gray731.png). :) Photo: Common octopus- albert kok/CC BY-SA (https://creativecommons.org/licenses/by-sa/3.0). Octopus nervous system: Jean-Pierre Bellier/CC BY-SA (https://creativecommons.org/licenses/by-sa/4 conditions an agent (or 'the brain') can discover the (amodal) notion of space in which it is embedded and infer its dimensionality from sensorimotor flow only. According to Piaget (1954), for the infant, initially, 'no constant relation exists between visual and buccal space or between tactile and visual space. True, auditory and visual space are already coordinated, as are buccal and tactile space, but no total and abstract space encompasses all the others'. Later, these spaces are connected through prehension (reaching and grasping), and the 'near' and 'far' spaces become differentiated. Eventually, through 'reversible operations'—e.g., whether the object moves in front of me or I move the head, the image on the retina will be the same—the child may overcome the space of individual modalities and 'objectify' the world, space, and its body; the body, say, will appear as an object with certain dimensions, independent of its perception by individual senses. Mathematical and algebraic tools and robotics can formalize Piaget's ideas and provide existence proofs for under what exact conditions an agent may develop spatial knowledge and give precise content to these concepts. Pioneered by Henri Poincaré, compensability (~ reversible operations) was exploited by Philipona, O'Regan, and Nadal (2003) who showed how an agent can infer the dimensionality of space from proprioception and exteroception, and this was extended by Terekhov and O'Regan (2016) to use coincidence detection in neural networks as the basis of a way of discovering the notion of space. Laflaquière, O'Regan, Argentieri, Gas, and Terekhov (2015) explicitly considered the agent's 'point of view' in the sensorimotor flows. An alternative approach makes use of self-contact; the 'body in space' can emerge from the structure of the proprioceptive–tactile space in self-touch configurations (Roschin, Frolov, Burnod, & Maier, 2011; Marcel, Argentieri, & Gas, 2016). Such models do not prove that these solutions are used by the brain, but they provide hypotheses and one can then look for them in the neural code.

Second, next to models at high level of abstraction like the simulated sensorimotor agents, robotics can provide a much higher degree of realism when it comes to mimicking biological bodies. This may be necessary to make progress beyond the existing models addressing coordinate transformations or multisensory integration (Xing & Andersen, 2000; Pouget, Deneve, & Duhamel, 2002) that typically concern very simplified scenarios with one- or two-dimensional geometry, one or two joint angles for the proprioceptive modality, etc. More than 15 years of research in the laboratory of Yasuo Kuniyoshi stands out in this respect; a highly realistic musculoskeletal fetal simulator (21 rigid body parts connected by 20 joints with 36 DoFs, 390 muscles with proprioceptive receptors, and 3000 tactile mechanoreceptor models) has been developed and coupled to a spinal circuit model (neural oscillators, α and γ motor neurons, and sensory interneurons) and a cortical model (2.6 million spiking neurons and 5.3 billion synaptic connections) (Yamada et al., 2016). Figure 10.8A shows the human-like distribution of tactile receptors on the fetal body. Mori and Kuniyoshi (2010) studied the effect of this distribution on the emergence of sensorimotor behaviours; with a natural (non-homogeneous) distribution, the fetus developed 'normal' kicking and jerking movements (i.e., similar to those observed in a human fetus), whereas with a homogeneous allocation, it did not develop any of these behaviours—just one illustration

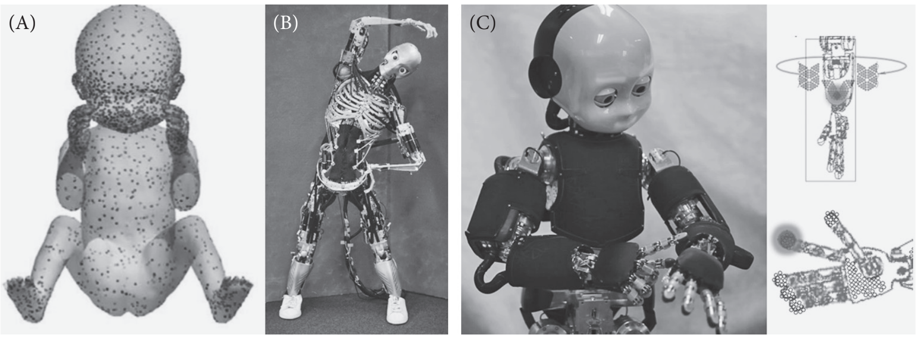


**Figure 10.8** Robots as embodied computational models of body representations. (A) Fetus simulator (Yamada et al., 2016). (B) Musculoskeletal robot Kenshiro (Asano, Okada, & Inaba, 2019). (C) iCub humanoid robot.

*Image credits*: (A) Reproduced from Yamada, Y. et al. An Embodied Brain Model of the Human Foetus, *Sci Rep*, 6 (27893), Figure 1d, doi:10.1038/ srep27893 (2016) under the Creative Commons Attribution 4.0 International License CC BY 4.0 (http://creativecommons.org/licenses/by/4.0/) (B) Reproduced with permission from (Yuki Asano, Kei Okada & Masayuki Inaba, 2019) and courtesy of Yuki Asano and JSK robotics laboratory in the University of Tokyo.

of the importance of the need for embodying the models related to body representations. Simulating physics can be computationally heavy and there is always a risk that the simulator does not get certain properties right. Therefore, physical robots are an indispensable tool. Figure 10.8B shows one of a series musculoskeletal humanoids—Kenshiro (160 'muscles': 50 in the legs, 76 in the trunk, 12 in the shoulder, and 22 in the neck) (Asano, Okada, & Inaba, 2019). Such platforms provide the right challenge to model the impact of the details of the motor system on body representations and reaching behaviours. Richter et al. (2016) have combined a musculoskeletal robotics toolkit (Myorobotics) with a scalable neuromorphic computing platform (SpiNNaker) and demonstrated control of a musculoskeletal joint with a simulated cerebellum. Finally, the iCub baby humanoid robot (see Figure 10.8C) is my platform of choice for models of body representations. It lacks some of the biological details of the other platforms—its whole body skin has a uniform density of tactile receptors and it is driven by standard electric motors rather than by artificial muscles—but it is a very versatile platform with all the key sensory and motor capacities. For example, we showed how it can be used to learn its 'somatosensory homunculus' (Hoffmann, Straka, Farkaš, Vavrečka, & Metta, 2018) or self-calibrate using self-touch (Roncone et al., 2014) or self-touch and self-observation (Stepanova et al., 2019).

## 10.4 Which characteristics of biological body representations should robots take on board?

What properties that biological body representations manifest should robot models copy? One feature that is clearly desirable is adaptivity. Robot models need to be developed in the first place. Then, additional calibration procedures may be required for every robot exemplar and also after deployment on the factory shop floor. During operation, the robot is subject to wear and tear or other conditions might change, calling for additional calibration. All of these processes are costly and often require the intervention of professionals with specialized equipment and possibly suspending production. The 'body in the brain', on the other hand, seems to develop largely from scratch and displays plasticity on all of these timescales—adapting to growth or failures, as well as performing rapid recalibration when working with a tool, for example. Automatic robot self-calibration is thus desired and solutions for this are being developed. For the robot to self-calibrate, it needs redundant sources of information about its body. There is a growing number of powerful, yet economic, sensors for robots available (cameras, RGB-D cameras, inertial measurement units, force/torque sensors, tactile sensors) and they can be exploited for calibration. Multimodality—another property of biological body representations—thus enables plasticity (see Figure 10.4). Such an extension of robot models should thus be unproblematic and is already happening.

I have sketched also other axes—robot body models are typically explicit, veridical (see Figure 10.5), universal, centralized, and modular (see Figure 10.7). All of these are—from an engineering perspective—very convenient properties. For example, being explicit and universal often implies that the models are capable of extrapolation; if transformations in three-dimensional space are represented using appropriate mathematical tools, they will always work—even in previously unseen circumstances. Implicit models would typically be expected to interpolate only, i.e., provide meaningful estimations within the range of existing examples only. Being veridical, or objective, implies that robot body models can be easily validated from the outside. The universal, centralized, and modular nature is ideal from a maintenance perspective. The kinematic model is only in one place and any updates will be automatically propagated to all other modules using it. One important additional convenient property that is a consequence of the features listed above is interpretability; it is possible to understand the model which is key for maintenance, debugging, etc. In this sense, there seem to be good reasons for preserving these properties.

Yet, these very characteristics are responsible for some inherent limitations. In particular, robot body models and associated control architectures lack robustness. The centralized and universal feature makes every module critical and that creates bottlenecks. Redundancy is against software development principles, but it importantly contributes to the resilience of animals. When faced with injuries, impairments, or lesions, they can find alternatives to performing a task. Implicit models are gaining popularity with the advent of deep learning (Nguyen et al., 2019) (for a survey in robotics, see, for example, Sünderhauf et al., 2018). This can be seen as a step toward brain-like models. Making robot control more embodied—exploiting the body morphology or local feedback loops—would be another step in this direction. However, there are trade-offs associated with this; mainly, the interpretability of such implicit, or black-box, models is reduced, which is a downside when they are part of applications where, for example, safety is at stake. Another difference is the sequential versus parallel processing (see Figure 10.6). Having multiple potential movement plans always ready will also improve the robustness of robot behaviour when faced with unexpected situations.

In summary, the strategy employed in robot modelling and control—a single, universal, veridical body model associated with corresponding control schemes—makes robots rather brittle when faced with failures or unexpected changes. Instead, the solution evolved by animals—multiple, distributed, partially overlapping, task-specific, and parallel architectures—makes them particularly robust and resilient.

## 10.5 Conclusion

I have outlined a number of characteristics of robot body models, as well as body representations in nervous systems. Most often, the nature of these models is very different, often ending up on opposite ends of different schematic axes. There seems to be a general trend—on many of the axes sketched in Section 10.3, the sequence is from robot body models, over body image and body schema, to the body representation in the octopus. Paraphrasing Brooks (1991), the octopus is most faithful to the strategy that *the body is its own best model*. Despite the efforts of Brooks and others, robots still heavily rely on models of their bodies—fixed, amodal, explicit, veridical, serial, centralized, and universal. Interestingly, the body image is the second closest to the robot side, which begs the question—asked by Yochai Ataria—to what extent the robots may be like Ian Waterman, the famous 'man who lost his body' (proprioception and touch) (Cole & Paillard, 1995). Indeed, just like deafferented subjects who 'lost their body schema', robots rarely recruit an implicit sensorimotor representation of their body or the body directly (without modelling it) (see also Hoffmann & Müller, 2017). Also, as their body models are explicit and veridical, they resemble the 'pictorial', body image-like representation of their bodies. Vision is also the main sensory modality and most self-calibration approaches rely on it. One difference remains—robots have proprioception, joint angle readings from encoders, and these, together with vision, are employed to bring about reaching movements. However, they rarely have touch (and may be clumsy and slow, just like Ian Waterman) and without their body model cannot do pretty much anything.

We have seen that the way animals and machines represent their bodies is quite different. Can robots contribute to our understanding of the 'body in the brain'? As we have seen in Section 10.4, this can be the case in two ways: (1) employing the mathematical machinery can provide proofs of what is possible—extracting body as an object in three dimensions from multimodal sensory information, for example; and (2) using robots as embodied computational models of body representations. In all cases, one should reflect upon using robots, such that the 'design decisions' typical for robot models are not blindly applied to the biological world. In terms of performance (Section 10.5), interestingly, the rather 'messy' way adopted by biology—which is also why our understanding of the mechanisms is still limited—is surprisingly good. Neither the models nor the behavioural performance are optimal, but they are good enough and highly robust. On the other hand, robot body models and control are very neat, transparent, universal, and overall highly parsimonious and optimized. Yet, the performance is fragile. One factor to consider here is that the criterion for quality of the models in animals is the success in the action/task for which the particular body representations were recruited. Robot body models ultimately also serve actions; however, the criterion for model quality is typically rather that it is a veridical representation of something—the robot link length, for example. The deep learning type of models may provide one of the ways of bridging machine learning and neuroscience (e.g., Richards et al., 2019).

## Acknowledgements

This work was supported by the Czech Science Foundation (GA CR), project EXPRO (nr. 20-24186X). The chapter draws partly on an earlier manuscript in preparation and discussions with Julius Verrel in 2012. I would like to thank the editors Yochai Ataria, Shogo Tanaka, and Shaun Gallagher for their comments and patience. I am also indebted to Kevin O'Regan, Alban Laflaquiere, and Christian Mangione for comments on a draft of the manuscript, and Gregor Schöner for discussions on the development of reaching. Yochai Ataria drew my attention to the parallel between robots and Ian Waterman. Kevin O'Regan pointed out the 'body in the brain' versus 'brain in the body' implications.

## References

Alsmith, A. J. & de Vignemont, F. (2012). Embodying the mind and representing the body. *Review of Philosophy and Psychology*, *3*(1), 1–13.

Azañón, E., Tamè, L., Maravita, A., Linkenauger, S. A., Ferrè, E. R., Tajadura-Jiménez, A., & Longo, M. R. (2016). Multimodal contributions to body representation. *Multisensory Research*, *29*(6–7), 635–661.

Asano, Y., Okada, K., & Inaba, M. (2019). Musculoskeletal design, control, and application of human mimetic humanoid Kenshiro. *Bioinspiration & Biomimetics*, *14*(3). IOP Publishing Ltd.

Berlucchi, G. & Aglioti, S. (2010). The body in the brain revisited. *Experimental Brain Research*, *200*(1), 25–35.

Bongard, J., Zykov, V., & Lipson, H. (2006). Resilient machines through continuous self-modeling. *Science*, *314*(5802), 1118–1121.

Botvinick, M. & Cohen, J. (1998). Rubber hands 'feel' touch that eyes see. *Nature*, *391*(6669), 756.

Brooks, R. A. (1991). Intelligence without reason. *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, 1, 569–595.

Carruthers, G. (2008). Types of body representation and the sense of embodiment. *Consciousness and Cognition*, *17* (4), 1302–1316.

Chinn, L. K., Hoffmann, M., Leed, J. E., & Lockman, J. J. (2019). Reaching with one arm to the other: coordinating touch, proprioception, and action during infancy. *Journal of Experimental Child Psychology*, *183*, 19–32.

Cisek, P. & Kalaska, J. F. (2003). Reaching movements: implications for computational models. In Michael A. Arbib (Eds.), *Handbook of Brain Theory and Neural Networks* (pp. 945–948). Cambridge, MA: MIT Press.

Cohen, Y. E. & Andersen, R. A. (2002). A common reference frame for movement plans in the posterior parietal cortex. *Nature Reviews Neuroscience*, *3*(7), 553–562.

Cole, J. & Paillard, J. (1995). Living without touch and peripheral information about body position and movement: studies with deafferented subjects. In J. L. Bermúdez, A. J. Marcel, & E. N. (eds.), *The Body and the Self* (pp. 245–266). Cambridge, MA: MIT Press.

Corbetta, D., Wiener, R. F., Thurman, S. L., & McMahon, E. (2018). The embodied origins of infant reaching: implications for the emergence of eye-hand coordination. *Kinesiology Review*, *7*(1), 10–17.

de Vignemont, F. (2010). Body schema and body image—pros and cons. *Neuropsychologia*, *48*(3), 669–680.

de Vignemont, F. (2015). Bodily awareness. *Stanford Encyclopedia of Philosophy*. Available from: https://plato.stanford.edu/entries/bodily-awareness/

Dijkerman, H. C. & De Haan, E. H. (2007). Somatosensory processing subserving perception and action: dissociations, interactions, and integration. *Behavioral and Brain Sciences*, *30*(2), 224–230.

Engel, A. K., Maye, A., Kurthen, M., & König, P. (2013). Where's the action? The pragmatic turn in cognitive science. *Trends in Cognitive Sciences*, *17*(5), 202–209.

Feldman, A. G. (2011). Space and time in the context of equilibrium-point theory. *Wiley Interdisciplinary Reviews: Cognitive Science*, *2*(3), 287–304.

Fuentes, C. T., Longo, M. R., & Haggard, P. (2013). Body image distortions in healthy adults. *Acta Psychologica*, *144*(2), 344–351.

Gallagher, S. (2006). *How the Body Shapes the Mind.* Oxford: Clarendon Press.

Georgopoulos, A. P., Kalaska, J. F., Caminiti, R., & Massey, J. T. (1982). On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex. *Journal of Neuroscience*, *2*(11), 1527–1537.

Graziano, M. (2006). The organization of behavioral repertoire in motor cortex. *Annual Review of Neuroscience*, *29*, 105–134.

Graziano, M. & Botvinick, M. (2002). How the brain represents the body: insights from neurophysiology and psychology. In W. Prinz & B. Hommel (eds.), *Common Mechanisms in Perception and Action: Attention and Performance* (pp. 136–157). Oxford: Oxford University Press.

Grush, R. (2004). The emulation theory of representation—motor control, imagery, and perception. *Behavioral and Brain Sciences*, *27*, 377–442.

Gutnick, T., Byrne, R. A., Hochner, B., & Kuba, M. (2011). Octopus vulgaris uses visual information to determine the location of its arm. *Current Biology*, *21*(6), 460–462.

Haugeland, J. (1985). *Artificial Intelligence: The Very Idea.* Cambridge, MA: MIT Press.

Hauschild, M., Mulliken, G. H., Fineman, I., Loeb, G. E., & Andersen, R. A. (2012). Cognitive signals for brain–machine interfaces in posterior parietal cortex include continuous 3D trajectory commands. *Proceedings of the National Academy of Sciences*, *109*(42), 17075–17080.

Head, H. & Holmes, H. (1911). Sensory disturbances from cerebral lesions. *Brain*, *34*, 102–254.

Heed, T., Buchholz, V. N., Engel, A. K., & Röder, B. (2015). Tactile remapping: from coordinate transformation to integration in sensorimotor processing. *Trends in Cognitive Sciences*, *19*(5), 251–258.

Hoffmann, M. & Müller, V. C. (2017). Simple or complex bodies? Trade-offs in exploiting body morphology for control. In G. Dodig-Crnkovic & R. Giovagnoli (eds.), *Representation and Reality in Humans, Other Living Organisms and Intelligent Machines* (pp. 335–345). Cham: Springer.

Hoffmann, M. & Pfeifer, R. (2018). Robots as powerful allies for the study of embodied cognition from the bottom up. In L. de Bruin, A. Newen, & S. Gallagher (eds.), *The Oxford Handbook 4e Cognition* (pp. 841–862). Oxford: Oxford University Press.

Hoffmann, M., Chinn, L. K., Somogyi, E., Heed, T., Fagard, J., Lockman, J. J., & O'Regan, J. K. (2017). Development of reaching to the body in early infancy: From experiments to robotic models. Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob), (pp. 112–119).

Hoffmann, M., Marques, H., Hernandez Arieta, A., Sumioka, H., Lungarella, M., & Pfeifer, R. (2010). Body schema in robotics: a review. *IEEE Transactions on Autonomous Mental Development*, *2*(4), 304–324.

Hoffmann, M., Straka, Z., Farkaš, I., Vavrečka, M., & Metta, G. (2018). Robotic homunculus: learning of artificial skin representation in a humanoid robot motivated by primary somatosensory cortex. *IEEE Transactions on Cognitive and Developmental Systems*, *10*(2), 163–176.

Holmes, N. P. & Spence, C. (2004). The body schema and the multisensory representation(s) of peripersonal space. *Cognitive Process*, *5*(2), 94–105.

Laflaquière, A., O'Regan, J. K., Argentieri, S., Gas, B., & Terekhov, A. V. (2015). Learning agent's spatial configuration from sensorimotor invariants. *Robotics and Autonomous Systems*, *71*, 49–59.

Leed, J. E., Chinn, L. K., & Lockman, J. J. (2019). Reaching to the self: the development of infants' ability to localize targets on the body. *Psychological Science*, *30* (7), 1063–1073.

Lenggenhager, B., Tadi, T., Metzinger, T., & Blanke, O. (2007). Video ergo sum: manipulating bodily self-consciousness. *Science*, *317*, 1096–1099.

Leyton, A. & Sherrington, C. (1977). Observations on the excitable cortex of the chimpanzee, orangutan, and gorilla. *Quarterly Journal of Experimental Physiology*, *11*(2), 135–222.

Lisman, J. (2015). The challenge of understanding the brain: where we stand in 2015. *Neuron*, *86* (4), 864–882.

Longo, M. R. (2015). Implicit and explicit body representations. *European Psychologist*, *20*(1), 6–15. doi:10.1027/1016-9040/a000198

Longo, M. R., Azañón, E., & Haggard, P. (2010). More than skin deep: body representation beyond primary somatosensory cortex. *Neuropsychologia*, *48*(3), 655–668.

Marcel, V., Argentieri, S., & Gas, B. (2016). Building a sensorimotor representation of a naive agent's tactile space. *IEEE Transactions on Cognitive and Developmental Systems*, *9*(2), 141–152.

Marder, E. (2007). Searching for insight: using invertebrate nervous systems to illuminate fundamental principles in neuroscience. In G. North & G. R.J. (eds.), *Invertebrate Nervous Systems* (pp. 1–18). Cold Spring Harbor: Laboratory Press.

Martin, V., Reimann, H., & Schöner, G. (2019). A process account of the uncontrolled manifold structure of joint space variance in pointing movements. *Biological Cybernetics*, *113*(3), 293–307.

Merleau-Ponty, M. (1945). *Phénoménologie de la Perception*. Paris: Gallimard.

Metta, G., Natale, L., Nori, F., Sandini, G., Vernon, D., Fadiga, L., … Montesano, L. (2010). The iCub humanoid robot: an open-systems platform for research in cognitive development. *Neural Networks*, *23*(8–9), 1125–1134.

Milner, D. & Goodale, M. (2006). *The Visual Brain in Action*. Oxford: Oxford University Press.

Mori, H. & Kuniyoshi, Y. (2010). A human fetus development simulation: self-organization of behaviors through tactile sensation. 2010 IEEE 9th International Conference on Development and Learning (ICDL). Ann Arbor, MI.

Neisser, U. (1988). Five kinds of self-knowledge. *Philosophical Psychology*, *1*(1), 35–59.

Nguyen, P. D., Hoffmann, M., Pattacini, U., & Metta, G. (2019). Reaching development through visuo-proprioceptive-tactile integration on a humanoid robot-a deep learning approach. Joint IEEE 9th International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob) (pp. 163–170). IEEE.

Nori, F., Traversaro, S., Eljaik, J., Romano, F., Del Prete, A., & Pucci, D. (2015). iCub whole-body control through force regulation on rigid non-coplanar contacts. *Frontiers in Robotics and AI*, *2*, 6.

Paillard, J. (1999). Body schema and body image—a double dissociation. In G. N. Gantchev, S. Mori, & J. Massion (eds.), *Motor Control, Today and Tomorrow* (pp. 197–214). Sofia: Prof. Marius Drinov Academic Publishing House.

Parmiggiani, A., Maggiali, M., Natale, L., Nori, F., Schmitz, A., Tsagarakis, N., … Metta, G. (2012). The design of the iCub humanoid robot. *International Journal of Humanoid Robotics*, *9*(4), 1250027.

Pattacini, U., Nori, F., Natale, L., Metta, G., & Sandini, G. (2010). An experimental evaluation of a novel minimum-jerk Cartesian controller for humanoid robots. Proc. IEEE/RSJ Int. Conf. Int. Robots and Systems (IROS). Taipei.

Penfield, W. & Boldrey, E. (1937). Somatic motor and sensory representation in the cerebral cortex of man as studied by electrical stimulation. *Brain*, *37*, 389–443.

Philipona, D., O'Regan, J. K., & Nadal, J.-P. (2003). Is there something out there? Inferring space from sensorimotor dependencies. *Neural Computation*, *15*, 2029–2049.

Piaget, J. (1954). *The Construction of Reality in the Child*. New York, NY: Basic Books.

Pitron, V., Alsmith, A., & de Vignemont, F. (2018). How do the body schema and the body image interact? *Consciousness and Cognition*, *65*, 352–358.

Pouget, A., Deneve, S., & Duhamel, J.-R. (2002). A computational perspective on the neural basis of multisensory spatial representations. *Nature Review Neuroscience*, *3*, 741–747.

Ramachandran, V. S. & Blakeslee, S. (1998). *Phantoms in the Brain: Human Nature and the Architecture of the Mind*. London, UK: Fourth Estate.

Richards, B., Lillicrap, T., Beaudoin, P., Bengio, Y., Bogacz, R., Christensen, A., . . . Gillon, C. (2019). A deep learning framework for neuroscience. *Nature Neuroscience*, *22*(11), 1761–1770.

Richter, C., Jentzsch, S., Hostettler, R., Garrido, J., Ros, E., Knoll, A., ... Conradt, J. (2016). Musculoskeletal robots: scalability in neural control. *IEEE Robotics and Automation Magazine*, *23*(4), 128–137.

Roncone, A., Hoffmann, M., Pattacini, U., & Metta, G. (2014). Automatic kinematic chain calibration using artificial skin: self-touch in the iCub humanoid robot. Proc. IEEE Int. Conf. Robotics and Automation (ICRA). Hong-Kong.

Roschin, V. Y., Frolov, A. A., Burnod, Y., & Maier, M. A. (2011). A neural network model for the acquisition of a spatial body scheme through sensorimotor interaction. *Neural Computation*, *23*(7), 1821–1834.

Scholz, J. P. & Schöner, G. (1999). The uncontrolled manifold concept: identifying control variables for a functional task. *Experimental Brain Research*, *126*(3), 289–306.

Schöner, G., Tekülve, J., & Zibner, S. (2018). Reaching for objects: a neural process account in a developmental perspective. In D. Corbetta & M. Sanntello (eds.), *Reach-To-Grasp Behavior: Brain, Behavior, and Modelling across the Life Span* (pp. 281–318). New York and London: Routledge.

Schwoebel, J. & Coslett, H. B. (2005). Evidence for multiple, distinct representations of the human body. *Journal of Cognitive Neuroscience*, *17*(4), 543–553.

Scott, S. H. & Kalaska, J. F. (1997). Reaching movements with similar hand paths but different arm orientations. I. Activity of individual cells in motor cortex. *Journal of Neurophysiology*, *77*(2), 826–852.

Stepanova, K., Pajdla, T., & Hoffmann, M. (2019). Robot self-calibration using multiple kinematic chains—a simulation study on the iCub humanoid robot. *IEEE Robotics and Automation Letters*, *4*(2), 1900–1907.

Sturm, J., Plagemann, C., & Burgard, W. (2009). Body schema learning for robotic manipulators from visual self-perception. *Journal of Physiology-Paris*, *103*(3–5), 220–231.

Sünderhauf, N., Brock, O., Scheirer, W., Hadsell, R., Fox, D., Leitner, J., . . . Corke, P. (2018). The limits and potentials of deep learning for robotics. *International Journal of Robotics Research*, *37*(4–5), 405–420.

Tamè, L., Azañón, E., & Longo, M. R. (2019). A conceptual model of tactile processing across body features of size, shape, side, and spatial location. *Frontiers in Psychology*, *10*, 291.

Terekhov, A. V. & O'Regan, J. K. (2016). Space as an invention of active agents. *Frontiers in Robotics and AI*, *3*, 4.

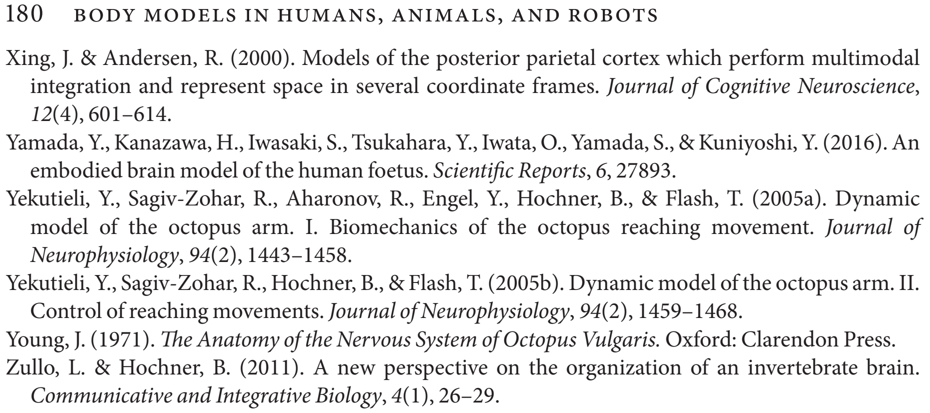
Vitzthum, H., Muller, M., & Homberg, U. (2002). Neurons of the central complex of the locust *Schistocerca gregaria* are sensitive to polarized light. *Journal of Neuroscience*, *22*, 1114–1125.

Walters, E. T., Bodnarova, M., Billy, A. J., Dulin, M. F., Díaz-Ríos, M., Miller, M. W., & Moroz, L. L. (2004). Somatotopic organization and functional properties of mechanosensory neurons expressing sensorin-A mRNA in *Aplysia californica*. *Journal of Comparative Neurology*, *471* (2), 219–240.

Webb, B. (2006). Transformation, encoding and representation. *Current Biology*, *16*(6), R184–R185.

Weber, D. J., London, B. M., Hokanson, J. A., Ayers, C. A., Gaunt, R. A., Torres, R. R., . . . Miller, L. E. (2011). Limb-state information encoded by peripheral and central somatosensory neurons: implications for an afferent interface. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *19*(5), 501–513.

Wong, A. M., Wang, J. W., & Axel, R. (2002). Spatial representation of the glomerular map in the *Drosophila protocerebrum*. *Cell*, *109*(2), 229–241.



Xing, J. & Andersen, R. (2000). Models of the posterior parietal cortex which perform multimodal integration and represent space in several coordinate frames. *Journal of Cognitive Neuroscience*, *12*(4), 601–614.

Yamada, Y., Kanazawa, H., Iwasaki, S., Tsukahara, Y., Iwata, O., Yamada, S., & Kuniyoshi, Y. (2016). An embodied brain model of the human foetus. *Scientific Reports*, *6*, 27893.

Yekutieli, Y., Sagiv-Zohar, R., Aharonov, R., Engel, Y., Hochner, B., & Flash, T. (2005a). Dynamic model of the octopus arm. I. Biomechanics of the octopus reaching movement. *Journal of Neurophysiology*, *94*(2), 1443–1458.

Yekutieli, Y., Sagiv-Zohar, R., Hochner, B., & Flash, T. (2005b). Dynamic model of the octopus arm. II. Control of reaching movements. *Journal of Neurophysiology*, *94*(2), 1459–1468.

Young, J. (1971). *The Anatomy of the Nervous System of Octopus Vulgaris.* Oxford: Clarendon Press.

Zullo, L. & Hochner, B. (2011). A new perspective on the organization of an invertebrate brain. *Communicative and Integrative Biology*, *4*(1), 26–29.

Appendix C

# Biologically inspired robot body models and self-calibration

Author's version of:

Hoffmann, M. (2021). Biologically inspired robot body models and self-calibration. In Ang, M., Khatib, O., and Siciliano, B., editors, *Encyclopedia of Robotics*. Springer-Verlag GmbH Germany, part of Springer Nature 2021.

DOI: https://doi.org/10.1007/978-3-642-41610-1_201-1

# Biologically inspired robot body models and self-calibration

Matej Hoffmann

## Synonyms

robot body schema, learning body representations, automatic self-contained robot calibration, robot model identification

## Definitions

body model refers to the representation of the robot body. Typical examples are robot kinematic model (e.g., using Denavit-Hartenberg convention) or robot dynamic model.

self-calibration is the process of automatically finding or fine-tuning model parameters by the robot itself—without the use of external metrology.

## Overview

Typically, mechanical design specifications provide the basis for a robot model and kinematic and dynamic mappings are constructed and remain fixed during operation. However, there are many sources of inaccuracies (e.g., assembly process, mechanical elasticity, friction). Furthermore, with the advent of collaborative, social, or soft robots, the stiffness of the materials and the precision of the manufactured parts drops and Computer-aided design (CAD) models provide a less accurate basis for the models. Humans, on the other hand, seamlessly control their complex

Matej Hoffmann
Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, Karlovo namesti 13 121 35 Praha 2, Czech Republic, e-mail: matej.hoffmann@fel.cvut.cz

bodies, adapt to growth or failures, and use tools. Exploiting multimodal sensory information plays a key part in these processes. In this chapter, differences between body representations in the brain and robot body models are established and the possibilities for learning robot models in biologically inspired ways are assessed.

When it comes to representation learning, the body is receiving little attention both in cognitive science and in Artificial Intelligence and deep learning: the focus is typically on world models or task learning. However, the body is not a mere interface between the mind or brain and the world, but plays an absolutely key part in behavior generation as well as cognition (e.g., (Pfeifer and Bongard 2007)). We can distinguish between the body model as such—referred to as body schema or body image in psychology and neuroscience—and forward and inverse models that link the body state with motor action and add the temporal dimension (see (Hoffmann et al 2010; Nguyen-Tuong and Peters 2011) for detailed accounts). Highly relevant are also mappings between joint space and task space (forward and inverse kinematics, plus differential kinematics) and between accelerations and forces (forward and inverse dynamics). Traditionally, robot model identification involves fine-tuning of parametric models using external metrology systems and non-linear least squares optimization (Hollerbach et al 2016). Mappings that are difficult to obtain analytically, such as inverse kinematics or dynamics, can be estimated using different regression methods ((Nguyen-Tuong and Peters 2011; Sigaud et al 2011) for surveys). Recently, deep learning applied to robotics (e.g., (Levine et al 2018)) has taken a completely different approach focusing on task-specific end-to-end mappings that largely sidestep body modeling. In line with this classification, an overview of three main research strands is presented: (i) automatic self-contained calibration where sensors on the robot and self-contact possibilities of humanoid robots are leveraged; (ii) biologically inspired body model adaptation; and (iii) deep learning approaches.

There are many ways in which one can characterize the properties of body models. Fig. 1 schematically illustrates some of them in two dimensions: adaptivity and “multisensoriality”. A number of prominent examples from robotics are shown, together with biological body representations in the brain. Traditionally, characteristics of body models in robotics and biology are largely opposite. However, with the advent of learning approaches in robotics, robot body models are catching up in some aspects. Finding the optimal nature of body models is a major challenge.

### *Adaptivity of body models*

Let us look at Fig. 1. The x-axis spans from fixed body models on the left to highly adaptive (also called plastic in neuroscience) models on the right. Traditional robot models are fixed: an industrial manipulator is shipped with its model. It may not be directly available to the customer, but it will be embedded in the robot controller which needs the robot model(s) for operation. The iCub humanoid robot (Metta et al 2010) and the kinematic model of its upper body is depicted in panel A. The model has been essentially hand-crafted by following the Denavit-Hartenberg (DH) con-

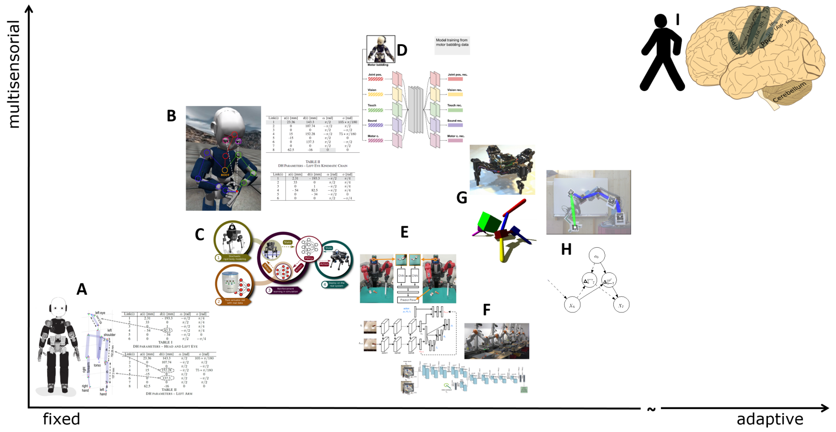


**Fig. 1** Body model characteristics. (A) iCub humanoid robot and its kinematic model. (B) iCub kinematic self-calibration using self-observation and self-touch (Stepanova et al 2019). (C) Hybrid model of the ANYmal robot (Hwangbo et al 2019). (D) Deep generative model of iCub sensorimotor space (Zambelli et al 2020). (E) Baxter robot poking – deep learning (Agrawal et al 2016). (F) Robot manipulators learning to grasp end-to-end (Levine et al 2018). (G) Four-legged machine learning its body structure (Bongard et al 2006). (H) Robot manipulator learning body structure from self-observation (Sturm et al 2009). (I) Human and schematic illustration of brain areas important for body representations. Figure revised and expanded from (Hoffmann 2021). Credit: A – iCub cartoon: Laura Taverna, Italian Institute of Technology. Credit: I – Walking human: Public domain (`https://commons.wikimedia.org/wiki/File:BSicon_WALK.svg`). Credit I – Brain image source: Hugh Guiney / Attribution-ShareAlike 3.0 Unported (CC BY- SA 3.0).

vention and supplying the corresponding lengths and angles from the CAD model of the robot. Body representations in primate brains (panel I) are located at the other extreme of the adaptivity axis. Neural representations, including those pertaining to the body are known for their plasticity on several time scales. First, body models need to be discovered by the brain, starting already in the fetal period. Second, body models need to adapt as the body grows. Third, they adapt or optimize when some body parts are frequently used in a specific task—when playing a musical instrument, for example. Fourth, they adapt also on very short time scales like when adapting to tools or when the subject is tricked by some of the well-known illusions, like the Rubber Hand illusion. Sometimes, the brain can be led into believing highly improbable things, like the "nose elongation" during Pinocchio illusion, but there are also limits to the plasticity. Important evidence suggestive of innate and fixed components of body models comes from the phantom limb phenomenon, which may be experienced even in subjects who congenitally lack limbs (e.g., (Ramachandran and Blakeslee 1998)). That is, the basic body layout may be to some extent hard-wired in the model and immutable.

Works in robotics feature different degrees and also types of adaptivity. The work of Stepanova et al (2019) (panel B) is an example of kinematic calibration of the complete robot body, including camera positions. The kinematic structure is known *a priori* and only the DH parameters of every link are calibrated. In case of panel C (Hwangbo et al 2019), the degree of adaptation is limited to learning the actuator dynamics: a mapping from motor commands to torques. The models in panels D (Zambelli et al 2020), E (Agrawal et al 2016), and F (Levine et al 2018) are all examples of deep learning. Zambelli et al (2020) use a variational autoencoder to enable the iCub to learn representations of its sensorimotor capabilities from different sensor modalities. This is a data-driven procedure and the generative model will essentially learn to reproduce "whatever is in the data streams". Embodiment constrains what these channels may contain and the body may thus be implicitly "represented" in the whole pipeline and in a somewhat compressed/abstract form in the hidden layers. However, no constraints on body layout, for example, are present. The examples in E (Agrawal et al 2016) and F (Levine et al 2018) are extensions of the traditional domain of Convolutional Neural Networks (CNNs), which is visual object recognition, to robot grasping or poking: a Cartesian space representation of the manipulation action is added to the images of the scene. Agrawal et al (2016) learn a prediction of the consequences of a poking action (forward model) as well as how to poke an object to achieve a desired effect (inverse model) within the same network and claim that this regularizes the feature space, forcing it into a more abstract representation than at pixel level. The network in (Levine et al 2018) predicts whether a particular grasp will be successful. Both architectures use end-to-end, model-free, learning and hence their degree of adaptivity is high in the sense that the mapping that will be learned is not engineered or constrained *a priori*. This, however, also means that what they learn is specific to the task at hand. They learn a representation related to success of a manipulation action in a given visual scene rather than a model of the robot body.

Bongard et al (2006) used a quadrupedal machine continuously "self-modeling" itself. The robot model had a special nature: it consisted of a physics-based simulator with a copy of the robot's limbs, motors, sensors and even the environment. Sturm et al (2009) had a robot arm observe "itself" using a camera and infer its model: learning the structure of a Bayesian network. These self-calibrating robots (panels G and H) are located most to the right from the works in robotics on the adaptivity axis, as they are able to learn any topology of their body layout (unlike B and C) and they are also able to adapt these reprentations online (unlike the other approaches and especially the deep learning models, C-F, where extensive offline training is typically required). Still, the machines' body models are importantly lagging behind the brain plasticity, as schematically illustrated by the break in the adaptivity axis on the right.

## *Multimodal nature*

The y-axis in Fig. 1, deals with the number of sensory (and motor) modalities that enter the model. Standard robot models are *amodal*—they do not depend on any sensory modality. They directly describe physical reality like the geometry of the body—the kinematic model of the iCub (panel A), for example. This holds in some sense also for many other robot body models: panels B, G, H. In B, the robot model itself is identical to that of A. The sensory modalities—proprioception, touch, and vision in this case—are needed to collect redundant information about the body's position in space and update the model. This layer is separated from the model of the robot geometry itself. In G, H, the situation is quite similar. In (Bongard et al 2006), the robot would search for its kinematic structure by comparing the actions and their sensory consequences (touch, tilt and clearance) from the physical world with those in the simulator. Sturm et al (2009) learn the structure of a Bayesian network from motor actions and observations in the camera. The models in E and F combine vision and some Cartesian vector related to action; the model in C maps motor commands to joint torques. The model of (Zambelli et al 2020) (panel D) with joint positions (proprioception), vision, touch, sound, and motor commands does not lag much behind the brain in terms of the number of modalities. In general, leaving aside how many modalities they use, the deep learning models (panels C, D, E, F) are different than the other robotic models (B, G, H) in that they do not merely use the sensory inputs to feed an amodal model. Instead, these neural network models intrinsically operate within the multimodal space.

Body representations in the brain (I) are famous for their multimodality. Azañón et al (2016) review the multisensory contribution to body representations: visual, somatosensory (tactile and proprioceptive), vestibular (inertial), auditory, and nociceptive (pain sensation). Hence, "body in the brain" scores highest on the multimodality axis. At the same time, body representations are assumed to be in some sense unified or coherent. In light of the works on robot self-calibration, this begs the question whether the brain arrives in some sense to an amodal, modality-independent, model of the, say, body in space, onto which different sensory modalities converge (see (Grush 2004)).

## *Universal and modular versus specialized end-to-end models*

There are also other axes that one can devise to investigate the character of body models. For example, neural representations are known to be distributed, while traditional robot representations are centralized. Whereas this "spatial aspect" may be also related to the computational substrate (neurons versus computers), more important is a functional division. Albeit centralized, robot body models are highly *modular*. For the iCub (Fig. 1A), there would normally be a single model of its kinematics and another one of its dynamics (mass distribution etc.). Then, there are distinct modules like forward/inverse kinematics and dynamics that may draw from the same robot model and be recruited for different purposes like state estimation, movement planning etc. There would be typically only one module of every kind (imagine a software library) providing this functionality upon request. The representations/modules will thus be *universal* and not overlapping. For deep learning applied to robotics, this is not the case. Levine et al (2018) or Agrawal et al (2016) specialize on a single task (grasping/poking); a different task will likely need a different network. The representations are thus end-to-end, task-specific and in case of multiple tasks also overlapping.

In nervous systems, there may also be complete sensorimotor loops specialized on indvidual tasks, partially overlapping or redundant. However, this approach does not scale well. In the human brain, there are areas where speficic representations of the body have been found. The somatotopic representations in the primary motor and somatosensory cortices (e.g. (Penfield and Boldrey 1937))—the "homunculi"—are well known. In the Posterior Parietal Cortex (PPC), the somatotopy is less pronounced, but it is regarded as a site where information about the body from different modalities converges. Specific areas related to representations of body parts or reaching targets in different reference frames have been found. These are recruited in different tasks or contexts and hence, there is certain universality and modularity. More details and visualizations are available in (Hoffmann 2021).

## Key Research Findings

The theory for traditional robot model identification is well established and determining kinematic as well as inertial parameters of robot manipulators is cast as least-squares parameter estimation (Hollerbach et al 2016). For robot control, additional mappings are often needed. Some can be readily obtained from the robot model (e.g., forward kinematics) while some (like inverse kinematics or dynamics) need to be approximated. A survey of different regression methods and their performance is provided in (Nguyen-Tuong and Peters 2011; Sigaud et al 2011). Here, a review of the following three research strands is provided: (i) automatic self-contained calibration, (ii) biologically inspired body model adaptation, and (iii) deep learning for robotics and its relation to body models.

### *Automatic self-contained calibration*

For humans and animals, there are no external measurement systems to help them calibrate their bodies and there is also no ground truth information regarding the geometric or inertial parameters of different body parts. Body models are developed in infancy in a self-contained multisensorial fashion—by matching the redundant information about the body contained in different sensory modalities. There is a growing number of powerful yet economic sensors for robots available (cameras, RGB-D cameras, inertial measurement units, force/torque sensors, tactile sensors) and they can be exploited for automatic self-contained calibration. Calibration of kinematic parameters is our focus here. The robot kinematic structure is typically represented using the DH convention or modified versions thereof (Hollerbach et al 2016)—all of which require only four rather than six parameters (3 translations, 3 rotations) to locate one reference frame relative to another. The DH convention, being more parsimonious, is advantageous as the number of parameters being calibrated is reduced. However, it only applies to structures consisting of revolute and prismatic joints. Sometimes, like for sensor locations on the robot body or custom end effectors, general 6D transformations may be needed. Alternative parametrizations like the twist representation used in (Martinez-Cantin et al 2010) are also possible. All the parameters of kinematic chain may be subject to calibration (e.g., (Hersch et al 2008; Stepanova et al 2019)) or only joint offsets, sometimes dubbed "daily calibration" (Nickels 2003) (e.g., (Vicente et al 2016)). Standard calibration proceeds in batch, i.e. offline, mode using non-linear least squares optimization methods like Gauss-Newton or Levenberg-Marquardt. Online methods, which are likely closer to how body models adapt in biological systems, have also been proposed (gradient descent for stochastic approximation (Hersch et al 2008); recursive least-squares estimation (Martinez-Cantin et al 2010); particle-based Bayesian estimation (Vicente et al 2016)).

### Self-observation

Cameras mounted on a robot can be used to calibrate its geometric parameters through self-observation of its end effectors or other parts of the body. This is a natural extension of open-loop kinematic calibration (Hollerbach et al 2016), whereby the external measurement system, like laser tracker, is replaced by a vision sensor on the robot itself. The theory for this approach is laid out in Bennett et al (1991) for a stereo camera system observing a robot arm, calibrating the manipulator's kinematics, extrinsic and intrinsic camera parameters. The main difference to open-loop calibration is that the location of the sensor will typically be calibrated as part of the robot kinematics—as a chain leading to the camera and terminating with camera extrinsic parameters—rather than a single 6D transformation from the robot reference frame to that of the external device. Sturm et al (2009) had a camera observe a manipulator with markers allowing 6D pose estimation on every robot link which provided enough information that even topology of the arm could be learned. More often, only the robot end effector can be observed and if a standard camera is used, robot pose cannot be observed in 3D space but only in camera reprojection (pixel space). The remainder of the works cited used humanoid robots. Recognition of the end effector can be facilitated by a colored marker (e.g., on a tool held by the robot (Hersch et al 2008); attached to the hand (Martinez-Cantin et al 2010; Nickels 2003)). Such fiducial markers can be avoided when the robot wrist, hand, or fingertip are identified directly in the image (Birbach et al 2015; Fanello et al 2014; Vicente et al 2016). Self-observation can be used to calibrate the robot kinematics, while camera parameters may be calibrated by other means and regarded as fixed (e.g., (Martinez-Cantin et al 2010)) or may be subject to calibration as well (extrinsic and intrinsic (Birbach et al 2015); extrinsic only (Stepanova et al 2019)). Fanello et al (2014), instead of tuning the robot parametric model, learned an additional transformation that can be applied on top of the kinematic and camera models and that compensated for residual errors in different positions in the workspace. Finally, self-observation can be also applied to learning models of higher order, such as velocity kinematics for the end effector (Droniou et al 2012).

### Self-contact

Self-contact constitutes a specific, less common, way of kinematic loop closure that is available only to humanoid-like or dual arm setups. Additionally, corresponding sensory and motor equipment such that this self-contact can be performed in a controlled manner is needed. One possibility is to utilize artificial electronic skins covering specific areas or complete robot bodies. A tactile array may be used for contact detection and if the skin itself is calibrated then also to measure additional components of the self-contact configuration—where contact occurs on each of two intersecting chains. Roncone et al (2014) performed kinematic calibration on the iCub using autonomous self-touch—index finger on the contralateral forearm; Li et al (2015) employed a dual arm setup with a sensorized "finger" and a tactile array on the other manipulator. Albini et al (2017) assumed the robot kinematics to be correct and used self-contact for spatial calibration of artificial skin. For robots without sensitive skins, if controlled self-contact can be established but the exact position not measured—typically when using force/torque sensing—such constraints can also be employed for calibration, similarly to closed-loop calibration (Hollerbach et al 2016).

### Inertial sensing

Inertial measurement units (IMUs; composed of 3-axis accelerometer, gyroscope, and magnetometer) constitute compact devices that many robots come already equipped with. Accelerometers are truly miniature and cheap and are often part of motor control boards distributed on the robot body, for example. Exploiting gravity projection into individual axes can be used as a convenient self-contained kinematic calibration method, avoiding the need for either self-observation or self-contact. Canepa et al (1994), moving one joint at a time, used a 3-axis accelerometer attached to the end effector of a 7 DoF robot arm to calibrate the axis orientation and center of rotation. To avoid numerical integration of accelerometer readings, D'Amore et al (2015) instead employed angular rate gyroscopes to identify the axis of rotation and used accelerometers without numerical integration. Mittendorfer and Cheng (2012) used accelerometers embedded in multimodal skin modules (Mittendorfer and Cheng 2011) to first perform "structural exploration" identifying which skin module belongs to which robot link. Second, kinematic parameters of a single kinematic chain were estimated. Finally, Guedelha et al (2016) using accelerometers in control boards on every link of the iCub humanoid proposed a procedure that does not require moving one joint at a time, but estimation is possible from slow movements of the complete kinematic chain.

### Multisensorial calibration

The full potential of automatic self-contained calibration is in simultaneous deployment of the different approaches discussed earlier and their optimal combination. While the brain seems to be good at (Bayes) optimal combination of multimodal sensory information (e.g., (Doya et al 2007)), multisensorial robot calibration has been rarely attempted. Using the multimodal skin (Mittendorfer and Cheng 2011), Mittendorfer and Cheng (2012) combined knowledge about the modules' dimensions, communication capabilities that allow the units to identify their neighbors and hence establish the skin topology, and accelerometer readings to reconstruct every sensors' pose. In (Mittendorfer et al 2014), this work is extended exploiting the LEDs also embedded in the modules: selectively turning on different modules' LEDs with different colors, salient visual features are formed. When observed by an external monocular camera, 6D pose of such "adaptive visual markers" can be estimated. Birbach et al (2015), using sensors in the head only, calibrated the humanoid robot Justin observing its wrist – see Fig. 2, left. Sensors were fused by minimizing a single cost function that aggregated errors obtained by comparing discrepancies between simulated projections (left and right camera images, Kinect image, Kinect disparity) and the wrist's position from forward kinematics. An inertial term from an IMU in the head was also considered. A self-calibration formulation combining self-observation and self-touch was proposed in (Hersch et al 2008). Stepanova et al (2019) systematically studied on the simulated iCub humanoid robot how self-observation, self-contact, and their combination can be used – see Fig. 2, right. They found that employing multiple kinematic chains ("self-observation" and "self-touch") is superior in terms of optimization results as well as observability. This approach was extended to a dual arm industrial robot in (Stepanova et al 2022).

## *Biologically inspired body model adaptation*

Inspiration from biology in learning and adapting body models can have very different forms. In (Bongard et al 2006; Cully et al 2015) the inspiration is at a very high, functional, level: animals are resilient and the walking robots in these works are as well: they can compensate for a broken motor, for example, and find an alternative gait. However, the body model used has little in common with nervous systems. Other capacities typical of biological body representations addressed by roboticists are tool use (e.g., (Nabeshima et al 2005)) or self-recognition (e.g., Gold and Scassellati (2009)). In works like (Fuke et al 2007) or (Yamada et al 2016), robots provide embodiment to models of the formation of body representations in the brain. Hoffmann et al (2010); Kuniyoshi (2019); Lanillos et al (2017); Schillaci et al (2016) provide surveys.

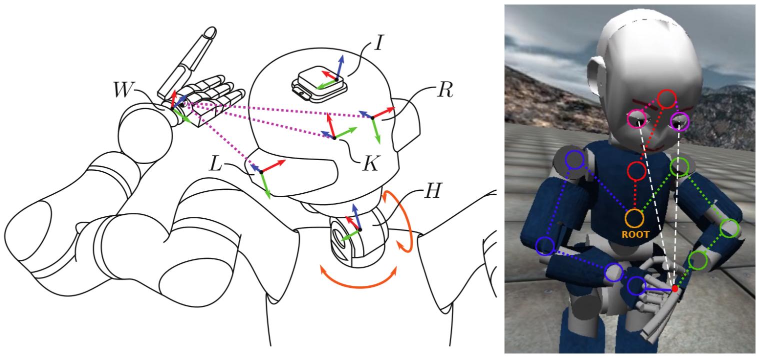


**Fig. 2** Multisensorial robot calibration. (Left) Sketch of calibration process of Agile Justin. The robot observes (magenta dotted lines) a point-feature attached to its wrist W with its two cameras and the Kinect sensor while moving the head (orange arrows). In addition, all measurements from the IMU, mounted in the robot's head, and the joint angle and torque sensors are recorded. This data together with the corresponding measurement models is fed into a least-squares estimator. The results are the calibrated poses of the left camera (L), Kinect sensor (K) and the IMU (I) relative to the head frame H, the cameras' spatial relationship (L relative to R), the cameras' and the Kinect's intrinsic parameters while considering joint angle offsets and arm joint elasticities. Figure and caption from Birbach et al (2015). (Right) Combining self-observation and self-contact. All kinematic chains originate in a common *Root* which is located at the third torso joint. The left and right arm chains are drawn in green and blue respectively. The eye chains have a common Root-to-head chain part marked in red. White lines denote projection into the eyes/cameras. Figure from Stepanova et al (2019).

## *Deep learning: where is the body model?*

Deep learning is finding its way into robotics, inheriting a strong bias toward visual perception and supervised, end-to-end, model-free solutions. Deep learning ((Goodfellow et al 2016) for an in-depth treatment) and its successes have been dominated by visual recognition problems, with convolutional neural networks (CNNs) as the key component. Typical problems involved image recognition and hence passive vision. Expanding the application domain to robotics implies bringing in the robot body and its actions in the environment—active perception and embodiment (Sünderhauf et al 2018). The robot body is almost never explicitly modeled, but it is inevitably implicitly present in many of the mappings being learned.

### Black-box modeling of residual errors

In some cases, deep neural networks are applied on top of traditional modeling and calibration approaches to compensate for remaining residual errors. These can be taken care of by traditional machine learning approaches (e.g., mixture of experts including Support Vector Machines and Gaussian processes for hand-eye calibration in (Fanello et al 2014)) or using deep neural networks (e.g., (Zhao et al 2019)). The work of Hwangbo et al (2019) (see also Fig. 1, C), constitutes a more sophisticated deployment of deep learning. An accurate simulator of the quadruped robot ANYmal is developed by combining classical rigid body modeling with a deep network for the actuator dynamics. In the simulator thus obtained, model-free deep reinforcement learning is used to learn policies—mapping of robot states to joint position targets—for different behaviors of the robot.

### End-to-end learning

End-to-end learning—training a learning system by applying gradient-based learning to the system as a whole—is at the core of the "deep learning revolution", as it avoids the need for hand-designing system components, sensory features etc. Beyond recognition and classification problems, it has been successfully applied to learning actions as well in so-called deep reinforcement learning. One learns a policy—a mapping from states to actions—for a particular task. In what follows, we are not concerned with abstract worlds like playing Go or Atari games, but focus on robots acting in the real world. Then, we review approaches focusing on learning input-output mappings related to the robot body rather than particular tasks.

**Learning tasks.** A straightforward example of applying deep learning to robotics is (Pfeiffer et al 2017). An expert operator is driving a mobile robot, providing training data for a CNN to learn a mapping from raw laser scanner data to steering commands for the robot. After learning, the robot can drive autonomously. In such cases, the teaching signal is available and deep learning architectures can be applied out of the box. In the work of Levine et al (2018) (see also Fig. 1, F), where end-to-end learning is applied to robot manipulation, the system learns successful grasps from experience. The CNN maps two input images of the scene—without and with the robot arm in the image—and a movement vector in Cartesian space into grasp success probability. The different movement vectors thus need to be sampled separately to find the best action. The network performs both object localization and gripper localization, subsuming both hand-eye coordination and grasp point detection. To map the high-level movement vector into targets for the robot joints, standard inverse kinematics is still used, however. A large amount of training data is needed: In the first experiment, about 800 000 grasp attempts were collected over the course of two months, using between 6 and 14 robotic manipulators at any given time. In the second experiment, 8 different robots were used to collect a dataset consisting of over 900 000 grasp attempts. Levine et al (2018) discuss that their method can be interepreted as a type of (deep) reinforcement learning. Overall, these architectures provide a solution for a single task. Learning is model-free and the consequence is limited generalization to other tasks.

The end-to-end approach can be augmented in several ways enforcing some regularization or model-building. Learning multiple tasks simultaneously is a natural extension. In (Singh et al 2019), a robot manipulator learns to arrange objects, place books, and drape cloth from a modest number of examples of successful outcomes, followed by actively solicited queries, where the robot shows the user a state and asks for a label to determine whether that state represents successful completion of the task. Agrawal et al (2016) achieve regularization of the learned mapping by simultaneously learning forward and inverse models. Imposing some modularity onto the network can also be beneficial: Andrychowicz et al (2020), dealing with in-hand object manipulation, separate the task into object pose estimation, learned with a CNN, and control policy learning using an LSTM (Long short-term memory) network. Finally, Lenz et al (2015) apply deep model predictive control to robot food cutting.

**Learning models.** While the majority of end-to-end approaches focuses on learning tasks, some works explicitly target learning models related to the robot body. Laflaquière and Hafner (2019) address robot visual self-recognition with a CNN composed of two branches: one predicting the image the robot will see with a given joint configuration and one estimating the error of this prediction in different parts of the image. The region with high predictability constitutes a mask for the robot's body in the image. Zambelli et al (2020) (see also Fig. 1, D) use a deep generative model—a variational autoencoder—to learn statistical relationships in the sensorimotor space of the iCub humanoid. The use of generative models and multimodal learning (Srivastava and Salakhutdinov 2014) constitute important directions that bring deep learning closer to body model learning. Learning a multimodal forward and inverse model linking the robot's perception and action in its full generality is, however, an immensely dificult task and we still do not fully understand how this is performed by the brain.

### Outlook: deep learning for robot body models

Hand in hand with the successes of deep learning, the community has also recognized the limitations. Lake et al (2017) contrast the state of the art in deep learning, which they dub "learning as pattern recognition", with the way humans learn and think: "learning as model building". End-to-end model-free supervised learning is typically limited to a single task and the system lacks robustness and flexibility. There are several approaches that channel learning in the direction of learning "good" representations (Goodfellow et al 2016, Chapter 15) like multi-task learning (Rahmatizadeh et al 2018), coding alternatives (Chang et al 2019), learning parameterized skills (Da Silva et al 2012), and transfer learning, which is not restricted to supervised learning tasks, but applicable to unsupervised and reinforcement learning as well. The way forward may be to guide learning such that features corresponding to underlying factors that appear in more than one setting are learned (Goodfellow et al 2016, Chapter 15). In the context of robotics, or physics in general, it remains to be seen whether higher layers of the networks will encode objects, general physical properties, forces, and approximately Newtonian dynamics (Lake et al 2017). The same applies to learning body models rather than world models. Unsupervised training or pre-training is likely to be most useful when the function to be learned is extremely complicated (Goodfellow et al 2016, Chapter 15), which certainly applies to learning the mappings related to the motors and sensors of the body. The brain cannot afford to have dedicated circuitry for every task. As the representations need to be shared, they become regularized and possibly modular, albeit in a different way than the classical engineering designs.

## Examples of Application

Although learning and automatic updating of robot body models is highly desirable, most of the existing approaches still have a proof-of-concept character and there are few real applications so far. The works of Hwangbo et al (2019) or Levine et al (2018), for example, are an exception—they are close to real practical deployment. However, the learned models are tuned for the concrete robot and environment and will likely catastrophically fail should the robot suffer some damage, for example. The works (Bongard et al 2006; Cully et al 2015), on the other hand, are specifically designed to cope with such situations. Yet, they are still rather "toy" than real robots.

## Future Directions for Research

Standard robot body models are explicit, veridical, accurate/optimal, universal, centralized, and modular (see Fig. 1 and (Hoffmann 2021) for additional details). All of these are from an engineering perspective very convenient properties. For example, being explicit and universal often implies that the models are capable of extrapolation; if transformations in 3D space are represented using appropriate mathematical tools, they will always work—even in previously unseen circumstances. Being veridical, or objective, implies that robot body models can be easily validated from the outside. The universal, centralized, and modular nature is ideal from a maintenance perspective. The kinematic model is only in one place and any updates will be automatically propagated to all other modules using it. One important additional convenient property that is a consequence of the features listed above is interpretability: it is possible to understand the model which is key for maintenance, debugging etc. In this sense, there seem to be good reasons for preserving these properties. Yet, these very characteristics are responsible for some inherent limitations. In particular, robot body models and associated control architectures lack robustness. The centralized and universal feature makes every module critical and that creates bottlenecks. Redundancy is against software development principles, but it importantly contributes to the resilience of animals.

Biological body models and control with multiple, distributed, partially overlapping, task-specific, and parallel architectures have largely opposite properties (Fig. 1, I). Up until ten years ago or so, brain-inspired body models were largely confined to the cognitive developmental robotics community ((Asada et al 2009; Hoffmann et al 2010) for surveys). Recently, with the advent of deep learning, there are models that parallel the brain adaptivity in some aspects and that are also closer to real applications. However, as discussed above, typically, one learns a task in a supervised end-to-end model-free manner, largely side-stepping learning of body models. The limitation of this approach is the specialization on a single task and lack of robustness in light of changing conditions.

One remedy to these shortcomings are hybrid models that marry the benefits of classical modular design with deep learning of some modules. The work of Hwangbo et al (2019), combining rigid body modeling with learning actuator dynamics and a mapping from state to action (policy), is a good example. Functionality that can be accurately addressed with traditional frameworks, like forward and inverse kinematics, can remain in use, while more difficult mappings like visual servoing and grasp point selection (Levine et al 2018) may be addressed using deep learning methods. A hybrid solution related to body models combines CNNs and a spatial, "white-box", model for body pose detection (Tompson et al 2014). Due to the origin of deep learning in visual perception, there has been little work on multisensorial learning (Srivastava and Salakhutdinov 2014). As robots come equipped with more and more powerful sensors, this aspect should certainly be incorporated in the future.

Alternatively, one can imagine a more radical brain-like approach. First, the result of neural (or neuromuscular) control are feasible rather than optimal solutions (Valero-Cuevas and Santello 2017). Second, there are typically multiple solutions to any task which can be selected on the fly. From an evolutionary perspective, having multiple good enough solutions at hand lends biological systems the versatility, adaptivity, and robustness. Here, it is important to consider how biological neural systems learn the models. There is evidence that different parts of the brain specialize on different learning types: cerebellum on supervised learning (mainly for motor control), basal ganglia on reinforcement learning, and the cortex on unsupervised learning (Doya 1999). Instead, the deep learning community is still heavily biased toward supervised learning. If we consider learning to reach in children, for example, it is still unclear how this ability is acquired. Herbort et al (2010) emphasize the unsupervised learning aspect and learn a task-independent body model, which encodes very general properties of the relationships between motor commands and body movements. A task-specific inverse model is extracted from this body model on the fly in the current context. To scale to the dimensionality of the motor system, the model of Ehrenfeld et al (2013) introduces modularity to the representation. Another hypothesis is that children are intrinsically motivated to learn this and other important skills (e.g., Oudeyer and Kaplan (2009)); there are many unstudied synergies between models of intrinsic motivation in developmental robotics and deep reinforcement learning systems (Oudeyer 2017). However, such developments are very far from current engineering practice. Intrinsically motivated, curious, robots that play to learn robust general skills would also require hardware that is similarly robust to what children have at their disposal. As new materials (e.g., McEvoy and Correll (2015)), soft robots Laschi et al (2016) and biohybrid robots Mazzolai and Laschi (2020) are gaining popularity, new, biologically inspired, methods for modeling and calibration will be the natural candidates to enable their control. Finally, with the bodies becoming more resilient and "intelligent", more control can be offloaded to them and the details of such interaction may not have to be modeled: paraphrasing Brooks (1991), the body may sometimes be its own best model.

## Acknowledgments

This work was supported by the Czech Science Foundation (GA CR), project no. 20-24186X.

## Cross-references

Camera and Hand-Eye Calibration, Cognitive Robotics, Developmental Robotics, Dynamic Models of Robots, Robot Self

## References

Agrawal P, Nair AV, Abbeel P, Malik J, Levine S (2016) Learning to poke by poking: Experiential learning of intuitive physics. In: Advances in neural information processing systems, pp 5074–5082

Albini A, Denei S, Cannata G (2017) Towards autonomous robotic skin spatial calibration: A framework based on vision and self-touch. In: Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on, IEEE, pp 153–159

Andrychowicz OM, Baker B, Chociej M, Jozefowicz R, McGrew B, Pachocki J, Petron A, Plappert M, Powell G, Ray A, et al (2020) Learning dexterous in-hand manipulation. The International Journal of Robotics Research 39(1):3–20

Asada M, Hosoda K, Kuniyoshi Y, Ishiguro H, Inui T, Yoshikawa Y, Ogino M, Yoshida C (2009) Cognitive developmental robotics: a survey. IEEE Transactions on Autonomous Mental Development 1(1):12–34

Azañón E, Tamè L, Maravita A, Linkenauger SA, Ferrè ER, Tajadura-Jiménez A, Longo MR (2016) Multimodal contributions to body representation. Multisensory research 29(6-7):635–661

Bennett D, Geiger D, Hollerbach J (1991) Autonomous robot calibration for hand-eye coordination. International Journal of Robotics Research 10 (5):550–559

Birbach O, Frese U, Bäuml B (2015) Rapid calibration of a multi-sensorial humanoids upper body: An automatic and self-contained approach. The International Journal of Robotics Research 34(4-5):420–436

Bongard J, Zykov V, Lipson H (2006) Resilient machines through continuous self-modeling. Science 314:1118–1121

Brooks RA (1991) Intelligence without reason. In: Proc. 12th Int. Joint Conf. on Artificial Intelligence

Canepa G, Hollerbach JM, Boelen AJ (1994) Kinematic calibration by means of a triaxial accelerometer. In: Proceedings of the 1994 IEEE International Conference on Robotics and Automation, IEEE, pp 2776–2782

Chang J, Kumar N, Hastings S, Gokaslan A, Romeres D, Jha D, Nikovski D, Konidaris G, Tellex S (2019) Learning deep parameterized skills from demonstration for re-targetable visuomotor control. arXiv preprint arXiv:191010628

Cully A, Clune J, Tarapore D, Mouret JB (2015) Robots that can adapt like animals. Nature 521(7553):503–507

Da Silva B, Konidaris G, Barto A (2012) Learning parameterized skills. In: International Conference on Machine Learning

D'Amore N, Ciarleglio C, Akin DL (2015) IMU-based manipulator kinematic identification. In: 2015 IEEE International Conference on Robotics and Automation (ICRA), IEEE, pp 1437–1441

Doya K (1999) What are the computations of the cerebellum, the basal ganglia and the cerebral cortex? Neural networks 12(7-8):961–974

Doya K, Ishii S, Pouget A, Rao RP (2007) Bayesian brain: Probabilistic approaches to neural coding. MIT Press, Cambridge, MA and London, England

Droniou A, Ivaldi S, Padois V, Sigaud O (2012) Autonomous online learning of velocity kinematics on the iCub: A comparative study. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, pp 3577–3582

Ehrenfeld S, Herbort O, Butz MV (2013) Modular neuron-based body estimation: maintaining consistency over different limbs, modalities, and frames of reference. Frontiers in Computational Neuroscience 7:148

Fanello SR, Pattacini U, Gori I, Tikhanoff V, Randazzo M, Roncone A, Odone F, Metta G (2014) 3d stereo estimation and fully automated learning of eye-hand coordination in humanoid robots. In: 2014 IEEE-RAS Int. Conf. on Humanoid Robots - HUMANOIDS '14

Fuke S, Ogino M, Asada M (2007) Body image constructed from motor and tactile images with visual information. International Journal of Humanoid Robotics 4(02):347–364

Gold K, Scassellati B (2009) Using probabilistic reasoning over time to self-recognize. Robotics and Autonomous Systems 57(4):384–392

Goodfellow I, Bengio Y, Courville A, Bengio Y (2016) Deep learning. MIT Press

Grush R (2004) The emulation theory of representation - motor control, imagery, and perception. Behavioral and Brain Sciences 27:377–442

Guedelha N, Kuppuswamy N, Traversaro S, Nori F (2016) Self-calibration of joint offsets for humanoid robots using accelerometer measurements. In: 2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids), IEEE, pp 1233–1238

Herbort O, Butz MV, Pedersen G (2010) The sure_reach model for motor learning and control of a redundant arm: from modeling human behavior to applications in robotics. In: From motor learning to interaction learning in robots, Springer, pp 85–106

Hersch M, Sauser E, Billard A (2008) Online learning of the body schema. International Journal of Humanoid Robotics 5:161–181

Hoffmann M (2021) Body models in humans, animals, and robots: mechanisms and plasticity. In: Ataria Y, Tanaka S, Gallagher  (eds) Body Schema and Body Image: New Directions, Oxford University Press, pp 152–180

Hoffmann M, Marques H, Hernandez Arieta A, Sumioka H, Lungarella M, Pfeifer R (2010) Body schema in robotics: A review. Autonomous Mental Development, IEEE Transactions on 2(4):304–324

Hollerbach J, Khalil W, Gautier M (2016) Model identification. In: Siciliano B, Khatib O (eds) Springer Handbook of Robotics, 2nd edn, Springer, pp 113–138

Hwangbo J, Lee J, Dosovitskiy A, Bellicoso D, Tsounis V, Koltun V, Hutter M (2019) Learning agile and dynamic motor skills for legged robots. Science Robotics 4(26):eaau5872

Kuniyoshi Y (2019) Fusing autonomy and sociality via embodied emergence and development of behaviour and cognition from fetal period. Philosophical Transactions of the Royal Society B 374(1771):20180,031

Laflaquière A, Hafner VV (2019) Self-supervised body image acquisition using a deep neural network for sensorimotor prediction. In: 2019 Joint IEEE 9th International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob), IEEE, pp 117–122

Lake BM, Ullman TD, Tenenbaum JB, Gershman SJ (2017) Building machines that learn and think like people. Behavioral and Brain Sciences 40:E253

Lanillos P, Dean-Leon E, Cheng G (2017) Enactive self: a study of engineering perspectives to obtain the sensorimotor self through enaction. In: 2017 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob), IEEE, pp 72–78

Laschi C, Mazzolai B, Cianchetti M (2016) Soft robotics: Technologies and systems pushing the boundaries of robot abilities. Science Robotics 1(1):eaah3690

Lenz I, Knepper RA, Saxena A (2015) Deepmpc: Learning deep latent features for model predictive control. In: Robotics: Science and Systems, Rome, Italy

Levine S, Pastor P, Krizhevsky A, Ibarz J, Quillen D (2018) Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. The International Journal of Robotics Research 37(4-5):421–436

Li Q, Haschke R, Ritter H (2015) Towards body schema learning using training data acquired by continuous self-touch. In: Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on, IEEE, pp 1109–1114

Martinez-Cantin R, Lopes M, Montesano L (2010) Body schema acquisition through active learning. In: Proc. IEEE Int. Conf. on Robotics and Automation (ICRA), pp 1860–1866

Mazzolai B, Laschi C (2020) A vision for future bioinspired and biohybrid robots. Science Robotics 5(38):eaba6893

McEvoy MA, Correll N (2015) Materials that couple sensing, actuation, computation, and communication. Science 347(6228)

Metta G, Natale L, Nori F, Sandini G, Vernon D, Fadiga L, von Hofsten C, Rosander K, Lopes M, Santos-Victor J, Bernardino A, Montesano L (2010) The iCub humanoid robot: An open-systems platform for research in cognitive development. Neural Networks 23(8-9):1125–1134

Mittendorfer P, Cheng G (2011) Humanoid multimodal tactile-sensing modules. Robotics, IEEE Transactions on 27(3):401–410

Mittendorfer P, Cheng G (2012) 3D surface reconstruction for robotic body parts with artificial skins. In: 2012 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp 4505–4510

Mittendorfer P, Cheng G (2012) Open-loop self-calibration of articulated robots with artificial skins. In: Robotics and Automation (ICRA), 2012 IEEE International Conference on, IEEE, pp 4539–4545

Mittendorfer P, Dean E, Cheng G (2014) 3D spatial self-organization of a modular artificial skin. In: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp 3969–3974

Nabeshima C, Lungarella M, Kuniyoshi Y (2005) Timing-based model of body schema adaptation and its role in perception and tool use: A robot case study. In: Proceedings. The 4th International Conference on Development and Learning, 2005, IEEE, pp 7–12

Nguyen-Tuong D, Peters J (2011) Model learning for robot control: a survey. Cognitive Processing 12(4):319–340

Nickels K (2003) Hand-Eye calibration for Robonaut. Tech. rep., NASA Summer Faculty Fellowship Program Final Report

Oudeyer PY (2017) Autonomous development and learning in artificial intelligence and robotics: Scaling up deep learning to human-like learning. Behavioral and Brain Sciences 40

Oudeyer PY, Kaplan F (2009) What is intrinsic motivation? a typology of computational approaches. Frontiers in neurorobotics 1:6

Penfield W, Boldrey E (1937) Somatic motor and sensory representation in the cerebral cortex of man as studied by electrical stimulation. Brain 37:389–443

Pfeifer R, Bongard JC (2007) How the body shapes the way we think: A new view of intelligence. MIT Press, Cambridge, MA

Pfeiffer M, Schaeuble M, Nieto J, Siegwart R, Cadena C (2017) From perception to decision: A data-driven approach to end-to-end motion planning for autonomous ground robots. In: 2017 IEEE International Conference on Robotics and Automation (ICRA), IEEE, pp 1527–1533

Rahmatizadeh R, Abolghasemi P, Bölöni L, Levine S (2018) Vision-based multi-task manipulation for inexpensive robots using end-to-end learning from demonstration. In: 2018 IEEE International Conference on Robotics and Automation (ICRA), IEEE, pp 3758–3765

Ramachandran VS, Blakeslee S (1998) Phantoms in the Brain: Human Nature and the Architecture of the Mind. Fourth Estate, London, England

Roncone A, Hoffmann M, Pattacini U, Metta G (2014) Automatic kinematic chain calibration using artificial skin: self-touch in the iCub humanoid robot. In: Robotics and Automation (ICRA), 2014 IEEE International Conference on, pp 2305–2312

Schillaci G, Hafner VV, Lara B (2016) Exploration behaviors, body representations, and simulation processes for the development of cognition in artificial agents. Frontiers in Robotics and AI 3:39

Sigaud O, Salaün C, Padois V (2011) On-line regression algorithms for learning mechanical models of robots: a survey. Robotics and Autonomous Systems 59(12):1115–1129

Singh A, Yang L, Hartikainen K, Finn C, Levine S (2019) End-to-end robotic reinforcement learning without reward engineering. arXiv preprint arXiv:190407854

Srivastava N, Salakhutdinov R (2014) Multimodal learning with deep boltzmann machines. The Journal of Machine Learning Research 15(1):2949–2980

Stepanova K, Pajdla T, Hoffmann M (2019) Robot self-calibration using multiple kinematic chains – a simulation study on the iCub humanoid robot. IEEE Robotics and Automation Letters 4(2):1900–1907

Stepanova K, Rozlivek J, Puciow F, Krsek P, Pajdla T, Hoffmann M (2022) Automatic self-contained calibration of an industrial dual-arm robot with cameras using self-contact, planar constraints, and self-observation. Robotics and Computer-Integrated Manufacturing 73:102,250

Sturm J, Plagemann C, Burgard W (2009) Body schema learning for robotic manipulators from visual self-perception. Journal of Physiology - Paris 103:220–231

Sünderhauf N, Brock O, Scheirer W, Hadsell R, Fox D, Leitner J, Upcroft B, Abbeel P, Burgard W, Milford M, et al (2018) The limits and potentials of deep learning for robotics. The International Journal of Robotics Research 37(4-5):405–420

Tompson JJ, Jain A, LeCun Y, Bregler C (2014) Joint training of a convolutional network and a graphical model for human pose estimation. In: Advances in neural information processing systems, pp 1799–1807

Valero-Cuevas FJ, Santello M (2017) On neuromechanical approaches for the study of biological and robotic grasp and manipulation. Journal of NeuroEngineering and Rehabilitation 14(1):101

Vicente P, Jamone L, Bernardino A (2016) Online body schema adaptation based on internal mental simulation and multisensory feedback. Frontiers in Robotics and AI 3:7

Yamada Y, Kanazawa H, Iwasaki S, Tsukahara Y, Iwata O, Yamada S, Kuniyoshi Y (2016) An embodied brain model of the human foetus. Scientific Reports 6:27,893

Zambelli M, Cully A, Demiris Y (2020) Multimodal representation models for prediction and control from partial information. Robotics and Autonomous Systems 123:103,312

Zhao G, Zhang P, Ma G, Xiao W (2019) System identification of the nonlinear residual errors of an industrial robot using massive measurements. Robotics and Computer-Integrated Manufacturing 59:104–114

# Appendix D

# Development of reaching to the body in early infancy: From experiments to robotic models

Authors' version of:

Hoffmann, M., Chinn, L. K., Somogyi, E., Heed, T., Fagard, J., Lockman, J. J. and O'Regan, J. K. (2017). Development of reaching to the body in early infancy: From experiments to robotic models. In *Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, pages 112-119.

DOI: `https://doi.org/10.1109/DEVLRN.2017.8329795`.

Author contributions: The contribution of M. Hoffmann was 30%.

# Development of reaching to the body in early infancy: From experiments to robotic models

Matej Hoffmann*†, Lisa K. Chinn‡, Eszter Somogyi§, Tobias Heed¶,
Jacqueline Fagard§, Jeffrey J. Lockman‡ and J. Kevin O'Regan§
*Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague
Email: matej.hoffmann@fel.cvut.cz
†iCub Facility, Istituto Italiano di Tecnologia, Via Morego 30, 16163 Genoa, Italy
‡Department of Psychology, Tulane University, New Orleans, LA, USA
§Laboratoire Psychologie de la Perception – CNRS UMR 8242, Université Paris Descartes, Paris, France
¶Biopsychology and Cognitive Neuroscience, Department of Psychology and Sports Science, and
Center of Excellence in Cognitive Interaction Technology (CITEC), Bielefeld University, Bielefeld, Germany

***Abstract*—We have been observing how infants between 3 and 21 months react when a vibrotactile stimulation (a buzzer) is applied to different parts of their bodies. Responses included in particular movement of the stimulated body part and successful reaching for and removal of the buzzer. Overall, there is a pronounced developmental progression from general to specific movement patterns, especially in the first year. In this article we review the series of studies we conducted and then focus on possible mechanisms that might explain what we observed. One possible mechanism might rely on the brain extracting "sensorimotor contingencies" linking motor actions and resulting sensory consequences. This account posits that infants are driven by intrinsic motivation that guides exploratory motor activity, at first generating random motor babbling with self-touch occurring spontaneously. Later goal-oriented motor behavior occurs, with self-touch as a possible effective tool to induce informative contingencies. We connect this sensorimotor view with a second possible account that appeals to the neuroscientific concepts of cortical maps and coordinate transformations. In this second account, the improvement of reaching precision is mediated by refinement of neuronal maps in primary sensory and motor cortices—the homunculi—as well as in frontal and parietal cortical regions dedicated to sensorimotor processing. We complement this theoretical account with modeling on a humanoid robot with artificial skin where we implemented reaching for tactile stimuli as well as learning the "somatosensory homunculi". We suggest that this account can be extended to reflect the driving role of sensorimotor contingencies in human development. In our conclusion we consider possible extensions of our current experiments which take account of predictions derived from both these kinds of models.**

## I. Introduction

The presence of various "body maps" in the brain has fascinated scientists and the general public alike, spurred by the account of Head and Holmes [1] and the discovery of somatotopic representations (the "homunculi") in the primary motor and somatosensory cortices [2]. The attention devoted to the representations of the body in the brain has also led to numerous attempts to describe or define them, and has given rise to proposals of a variety of concepts including superficial and postural schema [1], body schema, body image, corporeal schema, etc. Yet, these concepts are umbrella notions for a range of observed phenomena rather than the result of identification of specific mechanisms, and it has been criticized that this area of research is in a somewhat "chaotic state of affairs" [3], with limited convergence to a common view [4]. Here, we will focus on body representations that mediate implicit knowledge related to the body, its parts, and their posture relevant in the context of sensorimotor coordination.

It seems clear that body representations in the adult brain are a result of a complex interplay between genetic predispositions and both pre- and postnatal development. Work in recent years has focused on establishing the developmental trajectory of their underlying multisensory processes. This development starts in the fetus before birth (e.g., [5]; [6] for an embodied computational model), and then continues for many years (e.g., [7]–[9]). In this process, spontaneous movement and self-touch (which may also involve "grasping" the body) may play a key part. Infants frequently touch their bodies, with a rostro-caudal progression as they grow older—with head and trunk contacts more frequent in the beginning. As infants age, contacts become more caudal including hips, then legs, and eventually the feet [10]. The redundant information induced by these configurations in the motor-proprioceptive-tactile-visual manifold may facilitate learning about the body in space. Furthermore, as self-touch configurations are unique—with tactile stimulation on two different body parts and only in specific joint configurations—they might constitute a "contingent stimulus" associated with a reward or neuromodulation that bootstraps learning (e.g., [11]). As this knowledge develops, infants gain the ability to reach directly to targets on the body.

In the present article, we first report results from a series of completed as well as ongoing studies in which we observe infants' behavioral responses, including reaching and grasping, to stimulation with buzzers on different body parts (Section II). The results from these studies then provide constraints for a sensorimotor account of our observations (Section III), followed by a (brief) integration with evidence from the neurosciences (Section IV). Then we present our modeling endeavor on a humanoid robot with artificial sensitive skin (V) and close with a discussion (Section VI).

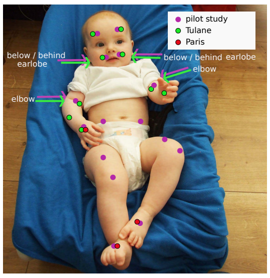


Fig. 1. An illustration of buzzer locations in each study reported here. Red dots indicate the cross sectional foot and hand locations study (Paris). Violet dots indicate the longitudinal pilot study. Green dots indicate the cross-sectional face and arms study (Tulane, New Orleans).

## II. Buzzers on infants' body and face

In two cross-sectional and one longitudinal study, we analyzed how 3 to 21 month old infants respond when a vibrating buzzer is attached to different parts of their bodies. We expected that differentiated movement patterns as a function of stimulus location would emerge only after about 4 months of age (c.f. [12]) and that well before infants are able to retrieve the buzzer [13], they would produce other behavioral responses that indicate knowledge of where their body was being stimulated.

### A. *Participants and Method*

In total we observed 122 infants of ages ranging from 3 to 21 months. The infants were supine, seated in an infant seat, or on the caregiver's lap. We attached a vibrating target to locations on the face and body using double-sided tape [13], a single location at a time and we left it there until either the infant had removed it or approximately 35 seconds had elapsed. The set of locations we used is shown in Fig. 1. From the video recordings we coded the infants' motor responses, in particular their overall limb activity and any (attempted or successful) reaches towards the buzzer. More details are available elsewhere [13], [14].

### B. *Which limb is it? Increased movement of specific body parts*

Our first cross-sectional study (Paris) involved 43 infants aged 3-6 months, where the buzzer locations were confined to the four limbs while the infants were in a fixed supine position in an infant seat. Very few infants were able to actually reach and grasp the buzzer, and their reactions mainly consisted of moving the limbs – see Fig. 2. We found that at 3 months, infants did not seem to differentiate stimulation on their different limbs, since independently of which limb was stimulated they responded to the buzzer stimulation in the same way, namely by increasing movements of the whole body. Interestingly, at 4 months, there was a global decrease in limb activity, and we did not find significant differences in limb activity across stimulation conditions. This may perhaps be explained by the fact that at this age, with the onset of reaching, the motor system is being reorganized and refined (e.g., [15]). From 5 months, infants demonstrated specific movement patterns associated with the stimulated hand or foot (moving the stimulated limb significantly more than the non-stimulated ones or touching the stimulated limb).

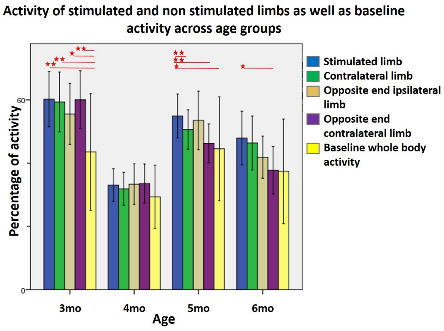


Fig. 2. Mean percentage of activity of the limb stimulated with a buzzer as compared to the three non-stimulated limbs and whole body activity at baseline (averaged activity of the four limbs) across age groups. In case of hand stimulations, 'Opposite end limb' refers to the feet and in case of foot stimulations it refers to the hands. Within each age group, significant differences between means are marked with an asterisk ($*p < 0.05; **p < 0.01$), as calculated with pairwise comparisons following the Generalized Estimating Equations (GEE) procedure.

### C. *Reaching for buzzers*

To study reaching towards and grasping the buzzer, we first conducted a pilot study with one child that was followed longitudinally at home from age 4 to 18 months and with a rich set of buzzer locations including the trunk and legs. In this study, posture was not fixed but alternated between sessions (mostly supine or infant car seat – collapsed in the Figure). Fig. 3 summarizes the results for our infant (up to 12 months). Each data point is the result for a single trial: "Contact" or "No Contact". We found that first successful manual buzzer contact was at the upper lip location – at 4.5 months. For the body locations, the abdomen, knee, and foot, success appeared between 5 and 6 months, with an apparent proximal-to-distal trend (the thigh location not following this pattern). Success for other locations on the face (forehead and below ears) appeared at around 7 months. Success for locations on the upper limb started with the hand after 6 months, and elbow locations only after 8 months.

We also conducted a cross-sectional study on 78 children aged 7 to 21 months to investigate reaching for the buzzer (Tulane, New Orleans). This time we only used buzzer locations on the face and arms. The results are summarized in Fig. 4. Similarly to the longitudinal pilot study, infants at 7 months could already reach to the mouth locations, but the ears and forehead developed more gradually. For the buzzer locations on the body (elbow, crook of elbow, forearm, palm, top of hand), infants at 7 months could already contact the hand buzzers, but the other buzzers emerged later. For the palm buzzer, 50 percent of buzzer contacts were made with the ipsilateral fingers, and 50 percent were made with the

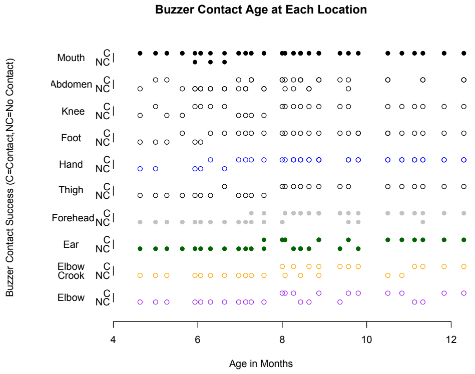


Fig. 3. Buzzer contact success (C=Contact, NC=No Contact) is shown for the pilot infant across age. The color code is chosen to match with the locations in Fig. 4 below, including filled markers for face locations. Each buzzer location is collapsed across left and right buzzer placement (hence the possibility of both C and NC for the same location and age).

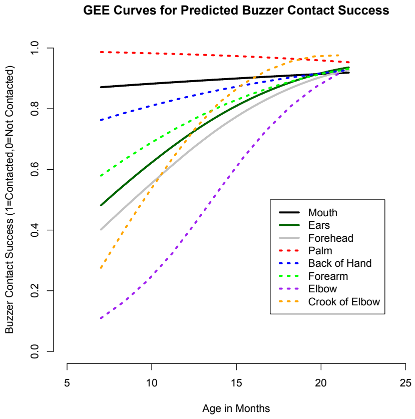


Fig. 4. The GEE predicted value of the average response is plotted over age for each buzzer location. Predicted responses were obtained using generalized estimation equations (GEE) testing the effects of Age, Buzzer Location, and the Age x Buzzer Location interaction on buzzer contact success. A value of 0 on the y-axis indicates an average prediction of no buzzer contact, while 1 indicates an average prediction of buzzer contact. Color codes match Fig. 3 for common locations.

contralateral hand. Contralateral reaches for the palm buzzer (and other locations that could be reached with either hand) increased with age. For the back of hand, all buzzer contacts were made with the contralateral hand, since ipsilateral contacts were impossible.

### D. *Conclusion of experiments with infants*

Overall, our infant experiment results suggest that: (i) at 3 and 4 months the infant does not respond to a buzzer on a limb by moving that particular limb, but rather responds in an undifferentiated way by moving its whole body; (ii) the limb-specific movement and buzzer-oriented reaching responses develop dramatically between 4 and 12 months. Certain locations are reached earlier than others, presumably because they correspond to innate reflexes (around the mouth), because they are easier to attain from the infant's natural postures, or because they do not move very much relative to the body.

## III. A SENSORIMOTOR FRAMEWORK

In order to understand the empirical data, we set out some theoretical suggestions. We divide our considerations into two parts: first, how does the infant's brain determine which body part is stimulated (to then possibly be able to move that body part)? Second, how does the infant's brain implement reaching to the buzzer with a hand or other effector?

### A. *Determining which body part is stimulated*

Evidence from the neurosciences suggests that somatosensory and motor "homunculi"—orderly neural maps that receive peripheral somatosensory input and, in turn, project from the cortex to the periphery—are present in the brain very early. However the mere presence of such map-like neuronal organization does not mean that the infant's brain "knows" where a stimulation is on its body, or how to move its arm rather than, say, its foot: the brain must establish mechanisms that link sensory and motor maps and allow the infant to make movements appropriate for the somatosensory input.

The basis of such mechanisms would presumably rely on the analysis of the statistical relationships of sensory and motor information, a mechanism that has been called "sensorimotor contingencies" [16]. The idea is that infant exploratory behavior may be guided by intrinsically defined rewards related to sensorimotor information (perhaps specifically related to tactile receptors [11] or "the joy of grasping" [17]), as articulated by the "intrinsic motivation" or "adaptive curiosity" frameworks [18]. During exploration, the infant's brain continually elicits actions and attempts to catalogue and organize the resulting sensory effects so as to become familiar with, and be able to predict, the resulting interactions it has with the world.

The existence of anatomically pre-wired maps in the brain, which approximately preserve the topology of the body, may be a starting point that facilitates the creation of coordinated movement and the organization of the related sensorimotor contingencies. Overall however, the task of extracting sensorimotor contingencies will be a difficult task because of the vast number of sensory inputs, motor outputs, and their statistical dependencies. We therefore expect this process of lining up sensory and motor information through their statistical properties to be long and gradual. In particular, we imagine that the correlations that will emerge most easily will be those that are most systematic. Thus, perhaps the simplest relationships to extract are those that link motor actions to immediately resulting proprioceptive changes: Whenever a muscle is innervated, accompanying muscle spindles will systematically tend to fire. Relations between motor output and tactile input will be almost as systematic. If the infant happens to move its hand, then not only proprioceptive sensors, but also the skin on the hand will deform, possibly brush against clothes or an object, and skin receptors on the hand will systematically signal a change; if the infant happens to move its foot, there will be systematic changes in the somatosensory foot region (we use somatosensory to denote proprioceptive and tactile afferents together).

Our sketch is simplistic in many ways. For instance, we currently ignore the question about the origin and nature of the initial movements. It is often assumed that infants initially move randomly ("motor babbling"), but more detailed accounts have been put forward of how behavior may emerge from basic neural properties [6] and changes in the role of transmitters and neuronal communication [19]. On the sensory side, we must consider that body surface areas that cannot be moved directly, like the back and abdomen, will also receive tactile stimulation through movements of the limbs: if the baby is lying on its back and pushes on a leg, stimulation on its back will change as the leg push causes the body to roll over.

Finally, our current sketch, as well as the tactile-motor experiments we have reported, ignore all social aspects of child development. Social interactions not only lead to numerous other-induced sensory experiences, but also embed these experiences in complex, reciprocal interaction cascades that are critical for the development of action concepts and language. Such factors, in addition to the basic sensorimotor processes we address in the current project, are most likely also important determinants of body representations.

Knowledge of sensorimotor contingencies does, furthermore, not per se explain which motor commands must be issued to attain a specific motor goal. Contingencies are developed as the sensory effects of ongoing movements, whereas a motor plan requires establishing the reverse contingencies, namely the motor commands necessary to obtain the related sensory signals. In psychology, ideomotor theory formalizes this approach and posits that motor actions are derived from the intended resulting sensory consequences [20]. In the current context, motor commands could be learned if the infant's intrinsic motivation mechanism includes the goal of increasing and refining its knowledge of particular sensorimotor correlations: this would cause the infant, given a sensory stimulation, to use motor babbling and later, systematic exploration, to discover the particular motor actions that give rise to that particular sensation. In this way it would learn, over time, which actions to undertake in order to obtain any particular sensation.

What would be the behavioural manifestation of the progressive learning of sensorimotor contingencies? If we assume that somatosensory and motor maps are, at first, only crudely organized, and that the learnt contingencies are initially unspecific, stimulation of a body location would not be expected to elicit movements of the baby specific to that location. On the contrary, we would expect fairly global—random or exploratory—motor responses of the whole body. But as knowledge of the contingencies improves in precision, the infant would progressively explore actions that more specifically modulate the sensation at a particular stimulated location. This is clearest in the case of stimulation of a limb: prior spontaneous movement of the limb might often produce tactile stimulation on that limb through rubbing of clothes or contact with an object. At a later time, when stimulated in a similar way on the limb, the infant's brain might try to explore how it could replicate or modulate that stimulation by moving the limb that was previously stimulated. To find the appropriate action, the infant would move the whole body, and then progressively narrow down its exploration until it finds the limb movement that most effectively modulates that stimulation.

We hypothesize that sensorimotor experience and extraction of contingencies are driving factors that guide the formation and refinement of unisensory and multisensory maps that relate body and environment. Presumably, these "body models" are strongly related to space in parietal regions, whereas the primary sensory and motor regions reflect statistics that are predominantly related to the physical body. Accordingly, we interpret the higher-level functions currently assigned to these areas, such as multisensory integration, memory, and executive control, as subfunctions related to the handling of the statistical properties of the organism's environment.

### *B. Reaching for a tactile stimulus*

Our proposal may provide an account for how a baby responds to touch on a limb, first by moving its whole body, and later by moving the appropriate limb. In this section, we apply this idea to our infant tactile stimulation experiments to attempt an explanation of the infant's hand reaches.

In the first stages of development, correlations between somatosensory and motor maps will allow the infant to move a limb that has been stimulated with a buzzer. More rarely however, and therefore learned presumably somewhat later in development, the infant will detect correlations deriving from (unintended) self-touch, that is, one of its hands touches some other body part, such as the trunk, the knee, the face, or the other hand. Neurons that are specifically related to self-touch have been reported in area 5 of the monkey parietal cortex [21].

Equipped with the intrinsic motivation to further explore such correlations, the infant will learn that, when stimulated say on the abdomen, it can recreate such stimulation by a certain motor command, namely moving one hand to that location. This then provides a mechanism by which the infant will eventually be able to reach towards a buzzer that is attached to that location.

However, such a mechanism will presumably provide movement trajectories only for previously used initial arm postures. We would expect that, at least at first, accurate reaching would only occur starting from these previously occurring arm postures. Furthermore, this simple explanation cannot explain how the infant might additionally account for the posture of the target limb. Body parts such as the leg, foot, or other hand can move in space relative to the trunk. Thus, information about the posture of the target limb must be factored into the movement.

It may be, as suggested by Graziano et al. [22] for monkeys, that such "factoring in" of a third, postural variable has been innately pre-wired for some behaviors in the form of "complex sensorimotor primitives". This would account for an infant's ability to reach to its mouth from any arm posture, and perhaps even independently of the head and mouth position. However, it is also conceivable that such movement patterns are learned in the womb, a place in which movement is strongly restricted, thus favoring certain particular contingency experiences. During development, these behaviors will be tuned as motor abilities unfold and the body changes shape. The tuning will be gradual and it will be based on learning of third-order correlations involving proprioception and/or vision, in addition to tactile and motor information.

The proposed mechanisms differ from the "classical" perspective based on vector geometry and transformation of spatial information between different reference frames [23]. Robotic simulations to be presented in Section V-C will help flesh them out.

## IV. Integrating sensorimotor and neuroscientific accounts

Over the last 30 years, neuroscientific research has established that many neurons in parietal and premotor cortex produce firing patterns that reflect spatial information in many different reference frames, integrating information from all sensory modalities. The prevalent view of sensorimotor processing in these regions is that this integration requires transformation between the reference frames inherent to the different senses, such as a 3D-like reference frame in vision and a skin-based reference frame in touch [23]. Infant development, then, would involve establishing and refining unisensory spatial maps and the transformations between them to allow sensory integration and their use for motor output. For instance, the infant must learn how tactile information on the foot must be combined with postural information of the leg to learn a transformation, or "mapping", from skin to 3D space. Having derived the 3D location of a touch by appropriate spatial transformation, an eye or hand movement can then be planned towards the spatial location of the tactile stimulus on the foot.

It is becoming increasingly obvious that these presumed transformation and integration processes optimise the use of information so as to be statistically optimal with respect to the reliability of each integrated signal (e.g. [24], [25]). In this view, development involves deriving the statistical properties of the body's sensory systems. For instance, the infant must learn that proprioceptive information of the leg is comparatively unreliable and learn to use vision to refine the estimate of limb position.

Several computational mechanisms that achieve transformation and optimal integration for some specific sensorimotor functions, such as integration of two senses, have been put forward. Current proposals involve different levels of abstraction, such as explaining behavior with Bayesian statistical principles and explaining neuronal firing in neural networks. Although they do not yet give a coherent view of the general mechanisms the brain may be using (see [23], [25]), such accounts are promising.

The sensorimotor contingency framework we have laid out above relies even more strongly on the idea that the brain derives statistical properties—those that link body and environment. Here, integrating sensory information, transforming between reference frames, and taking account of signal reliability, are by-products of the brain's main purpose of relating body and world for the purpose of action. In this view, the infant must learn that a concurrent touch on the left hand and the nose, combined with specific postural information of the arm, entails that the hand has touched the nose, and that this sensory information must be reproduced to relieve an itch on the nose at a later time.

These considerations represent interesting ideas for robot development: they suggest that both the modular structure of cortical regions and the characteristics of neuronal firing may be emergent properties of an overarching processing principle.

## V. Remapping and self-touch in robots

The body, with its geometry and sensorimotor capacities, is of course constitutive in the construction of body representations. Thus, simulated agents or, better, robots, constitute the best tools for such a modeling endeavor. For human body representations, humanoid robots are the platform of choice. Robot "body schemas" often have quite different characteristics (fixed, centralized, explicit, amodal) than what we expect from their biological counterpart, but there has been work on robot body models learning (self-calibration) as well as modeling the biological body representations and their development using robots (see [26], [27] for surveys). The majority of this work is more in line with the "classical" account of body representations (with explicit frame of reference transformations and perception separated from action), which lends itself more easily to robotic implementations. However, there are notable exceptions: in particular the work of Kuniyoshi and colleagues (e.g., [6]) dealing with fetal development.

### A. *The iCub humanoid robot with whole-body artificial skin*

The morphology of the iCub humanoid robot (Fig. 5B, [28]) is modeled after a 4-year old child: it has a similar kinematic configuration and sensory repertoire to humans (on some level of abstraction). Importantly, it has been recently equipped with a whole-body artificial skin comprising around 3000 pressure-sensitive tactile elements (taxels) (Fig. 7 A). Thus, it is now possible to model body representation acquisition through tactile-proprioceptive-visual-motor correlations on this platform. The parallel of a baby removing a buzzer and the robot performing self-touch on the torso is in Fig. 5.

### B. *Reaching for a tactile stimulus using inverse kinematics*

In Roncone et al. (2014) [29], we implemented reaching for a tactile stimulus by modifying classical robotic solutions and developing a new inverse kinematics formulation to deal with

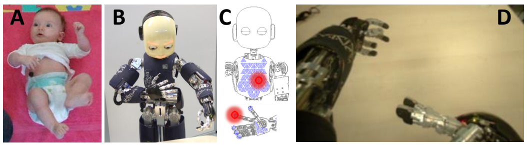


Fig. 5. Self-touch on torso. (A) Removal of buzzer from abdominal area in 5-month old infant. (B) iCub touching its trunk with index finger. (C) Tactile stimulation corresponding to "double touch" event—torso and right index finger. (D) View from iCub's left camera—contact location out of sight.

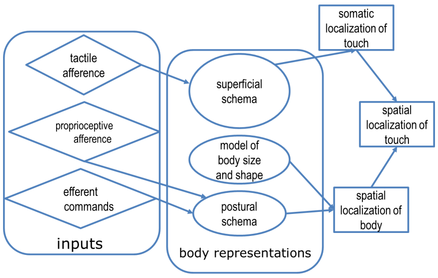


Fig. 6. Model of tactile spatial localization. Adapted from [30].

the fact that the target is on the body rather than fixed in space (cf. video: https://www.youtube.com/watch?v=pfse424t5mQ). This work serves as a baseline here: the behavior is instantiated in the robot, thus the whole loop from tactile stimulation to motor action is in place. Our next steps consist of replacing the engineered modules with those that are inspired by infant development (behaviorally and neurally).

### C. Remapping decomposed into modules

Longo et al. [30] propose that spatial localization of touch may be obtained by combining (1) tactile localization on skin and (2) spatial localization of body / position sense (i.e. where in space the particular body part is). These components (high-level percepts) draw on tactile and proprioceptive (and possibly motor) inputs and three modules or representations: superficial schema, postural schema, and a model of body size and shape – as illustrated in Fig. 6. This approach provides an easier starting point for robotic modeling than the sensorimotor approach. We have set out to implement this modular framework in the robot, which will be detailed in the next sections.

*1) Tactile homunculus (superficial schema):* One component or "representation" that seems necessary is the "tactile homunculus" or superficial schema. In Hoffmann et al. [31], we have obtained this homuncular representation for one half of the upper body of the iCub humanoid: Local stimulations of the skin surface were fed into a self-organizing map algorithm (SOM) that was additionally constrained such that the sequence of body parts on the output sheet mimicked that from the cortex (area 3b) – see Fig. 7B. This representation now provides a building block that we can deploy in further modeling.

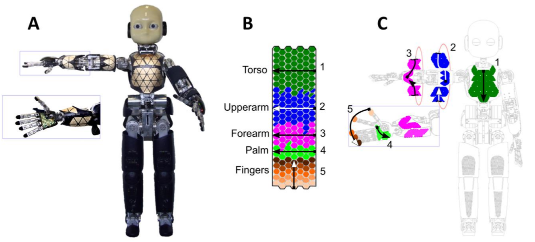


Fig. 7. iCub skin and tactile homunculus. (A) Photograph of the iCub robot with artificial skin exposed on the right half of the upper body (1154 taxels in total). (B) Representation of tactile inputs learned using a Self-Organizing Map – a 24 × 7 neuronal sheet. (C) Schematics with skin patches unfolded and colored to mark the correspondence with (B). Arrows illustrate the relationship in orientation between the skin parts and the learned map. From [31].

*2) Proprioceptive homunculus (postural schema):* In Hoffmann and Bednarova [32], we strived to provide a computational model of the representation of proprioception in the brain and its development. We used a simulator of the iCub humanoid robot and had it randomly move its arm in front of the face and follow the arm with gaze, thus "babbling" in Cartesian space, inspired by analogous behavior in infants ("hand regard"). Unlike for the "superficial schema", attempting to obtain a robot "postural schema" revealed a number of gaps in our knowledge.

As elaborated in [32] in more detail and with additional references, first, the principal proprioceptors are constituted by muscle spindles, which deliver information about muscle length and speed. In primates, this information is relayed to the primary somatosensory cortex and eventually the posterior parietal cortex, where integrated information about body posture (postural schema) is presumably available. However, it is not clear what variable neurons in the ascending pathway and in the cortex are actually encoding. To an engineer, joint angles would seem the most useful variables. However, the lengths of individual muscles have nonlinear relationships with the angles at joints and it is not clear where this transformation would occur. Second, Kim et al. (2015) [33] identified different types of proprioceptive neurons in SI, namely neurons that fire proportionally to joint angle (single or multi-digit) and those that directly register posture. The SOM algorithm seems to naturally support the latter type only though, learning to pick up the most frequent postures (or "postural synergies"). In summary, the nature of encoding of posture as well as the development of the postural schema remains unclear.

*3) Spatial localization of touch – a neural model proposal:* Finally, we present our work in progress—a proposal for a neurorobotic implementation of the scheme proposed in [30] (Section V-C and Fig. 6) – see Fig. 8. Tactile afferents are simply the pressure values read off the individual taxels stimulated on the robot skin. Proprioceptive afferents consist of joint angles of all the degrees of freedom (joints) relevant for the task. The primary representations draw on the output of previous sections. The activations in the superficial schema—

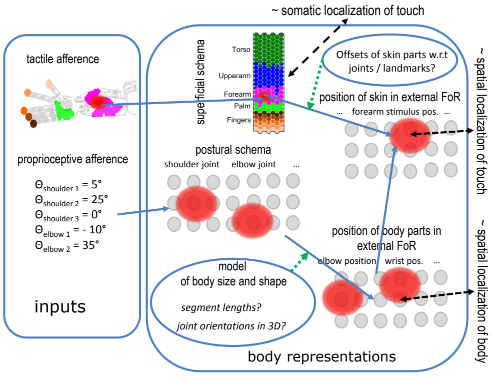

~ somatic localization of touch
tactile afference
proprioceptive afference
$\Theta_{\text{shoulder 1}} = 5°$
$\Theta_{\text{shoulder 2}} = 25°$
$\Theta_{\text{shoulder 3}} = 0°$
$\Theta_{\text{elbow 1}} = -10°$
$\Theta_{\text{elbow 2}} = 35°$
inputs
superficial schema
Torso
Upperarm
Forearm
Palm
Fingers
Offsets of skin parts w.r.t joints / landmarks?
position of skin in external FoR
... forearm stimulus pos. ...
postural schema
shoulder joint
elbow joint
...
position of body parts in external FoR
elbow position
wrist pos.
...
model of body size and shape
segment lengths?
joint orientations in 3D?
body representations
~ spatial localization of touch
~ spatial localization of body

Fig. 8. Spatial localization of touch – neural network model proposal (cf. Fig. 6). FoR = Frame of Reference. See text for details.

which is a distorted map of the skin surface—provide tactile localization on the skin. The key challenge is to determine the form that the "model of body size and shape" will take. As an initial hypothesis, we can consider this model as knowledge embedded in (synaptic) weights in the connections from the postural schema to the "position of body parts in external FoR" (FoR = Frame of Reference) neural population. First guesses as to what the information encoded may be are guided by the forward kinematics mapping in the robot (segment lengths and joint orientations in 3D). The "position of body parts in external FoR" hypothetical neural population thus corresponds to the spatial localization of body. Combining the inputs from the "superficial schema" and the "position of body parts in external FoR" populations finally gives rise to remapping of the tactile stimulus into external coordinates (it seems that knowledge about the position/offset of skin parts with respect to some joints/landmarks will also be needed).

To wrap up the section on robotic modeling, our work so far has a bias toward "classical", modular, divide-and-conquer solutions, as is often symptomatic of robotic and artificial intelligence approaches to cognition. In the future, we will strive to obtain the "modules" and the use of specific frames of reference in an emergent fashion—as proposed at the end of section IV. The primary homuncular-like representations obtained for the robot may then still constitute useful building blocks, but their connections and interactions should emerge in a more holistic sensorimotor setting. The (intrinsic) motivational component that may drive the exploration and facilitate learning constitutes one of our next foci.

## VI. Discussion, Conclusion, Future Work

The empirical data, conceptual framework, and robotic modeling described above are at present too disparate to allow a proper connection between them. The sensorimotor approach is not sufficiently developed to make precise predictions about the course of infants' responses to buzzers. Models using the humanoid robot are currently mostly couched in the classical framework in which "spatial remapping" to reach for the body occurs through an explicit chain of frame of reference transformations, rather than implicitly as presumably happens in the infant's brain. We are however planning work more in line with the sensorimotor account. Meanwhile our collaboration has generated new hypotheses, insights, and predictions that will be investigated further in existing data or tested in future experiments. Some of the points that have come to light are as follows.

There undoubtedly exist some (more or less complex) prewired primitive movement "reflexes" [22], that may help bootstrap the baby's acquisition of body knowledge. It will be interesting to check whether such reflexes already incorporate the equivalent of coordinate transformations, allowing, for example, a reach to the mouth to occur independently of the arm's starting position, and independently of the rest of the baby's posture. Primitive reflexes, and more generally the first reaching movements, may also be limited by the fact that connections between brain hemispheres develop later than connections within hemispheres. We are planning in future to confront this idea with the empirical data.

Another factor that may prioritize development of certain particular reaching capacities might be the baby's habitual resting posture, favouring natural frequent contact of the hands with certain body parts, and thereby learning of reaching to those parts. In particular, in our longitudinal pilot study with a single baby we observed that between 5 and 7 months of age, the infant's arms initially rested around the waist, but later, as the legs started bending, the hands would more often spontaneously contact the knees and then the feet—these habitual resting locations were then correlated with the progression of buzzer removal success. This is to be documented more accurately on a larger sample.

Presumably, but this remains to be confirmed in further analysis of the empirical data, reaching that necessitates remapping of spatial coordinates (for example to the other hand or to the legs) appears later than reaching to body parts that are more or less fixed with respect to the body (the abdomen, the face region – although this moves to some extent). Also, we expect that a reach learned starting from an initial arm posture will not be possible from a different arm posture. Evaluating these claims will be a way of distinguishing sensorimotor type models from the more classical models in which reaching is realized in a sequential fashion composed of stimulus localization, remapping into an external reference frame, and computation of a reaching trajectory to that location—whereby these transformations would have to automatically incorporate current body posture. Another point is the implication of vision. We have not yet included vision in our analyses—since it seems that at first infants are not using visual feedback. However clearly at the later stages of development visual tracking must be included, and we will modify the models that we postulate to accommodate the results.

Finally, we have been focusing on reaching with the upper limbs in order to understand how body maps develop. It should be noted, however, that the lower limbs are effectors as well and can be used to localize targets on the body, especially those that are situated on the opposite leg. Our informal observations suggest that young infants spend a considerable amount of time rubbing their feet together as well as rubbing one foot along the opposite leg. Infants also spend time bringing their feet to their hands (or vice versa) and in some instances bringing the foot to the face. Understanding how a fully integrated map of the body develops will also require viewing the feet and legs as effectors that are incorporated into an overall sensorimotor representation of the body.

In conclusion, our converging program of infant and robotic approaches has provided a starting point for further work on modelling how the infant reaches its body.

## Acknowledgment

M.H. was supported by a Marie Curie Intra European Fellowship (iCub Body Schema 625727) within the 7th European Community Framework Programme and the Czech Science Foundation under Project GA17-15697Y. E.S., J.F. and K. O'R. are grateful for financing from ERC Advanced Project #323674, FEEL and FET-OPEN grant GoalRobots. L.C. and J.L. were supported in part by the National Institutes of Health Award 5R01HD067581. T.H. is supported by Emmy Noether grant He 6368/1-1 by the German Research Foundation (DFG).

## References

[1] H. Head and H. G. Holmes, "Sensory disturbances from cerebral lesions," *Brain*, vol. 34, pp. 102–254, 1911.

[2] W. Penfield and E. Boldrey, "Somatic motor and sensory representation in the cerebral cortex of man as studied by electrical stimulation," *Brain*, vol. 37, pp. 389–443, 1937.

[3] G. Berlucchi and S. M. Aglioti, "The body in the brain revisited," *Exp. Brain Res.*, vol. 200, pp. 25–35, 2009.

[4] F. de Vignemont, "Body schema and body image - pros and cons," *Neuropsychologia*, vol. 48(3), pp. 669–680, 2010.

[5] M. Milh, A. Kaminska, C. Huon, A. Lapillonne, Y. Ben-Ari, and R. Khazipov, "Rapid cortical oscillations and early motor activity in premature human neonate," *Cerebral Cortex*, vol. 17, no. 7, pp. 1582–1594, 2006.

[6] Y. Yamada, H. Kanazawa, S. Iwasaki, Y. Tsukahara, O. Iwata, S. Yamada, and Y. Kuniyoshi, "An embodied brain model of the human foetus," *Scientific Reports*, vol. 6, 2016.

[7] A. J. Bremner and C. Spence, "Chapter seven - the development of tactile perception," in *Advances in Child Development and Behavior*, ser. JAI, 2017, vol. 52, pp. 227–268.

[8] M. Gori, M. Del Viva, G. Sandini, and D. C. Burr, "Young children do not integrate visual and haptic form information," *Current Biology*, vol. 18, no. 9, pp. 694–698, 2008.

[9] B. Pagel, T. Heed, and B. Röder, "Change of reference frame for tactile localization during child development," *Developmental science*, vol. 12, no. 6, pp. 929–937, 2009.

[10] B. L. Thomas, J. M. Karl, and I. Q. Whishaw, "Independent development of the reach and the grasp in spontaneous self-touching by human infants in the first 6 months," *Frontiers in psychology*, vol. 5, p. 1526, 2015.

[11] O. Sporns and G. M. Edelman, "Solving Bernstein's problem: A proposal for the development of coordinated movement by selection," *Child development*, vol. 64, no. 4, pp. 960–981, 1993.

[12] H. Watanabe and G. Taga, "General to specific development of movement patterns and memory for contingency between actions and events in young infants," *Infant Behavior and Development*, vol. 29, no. 3, pp. 402–422, 2006.

[13] J. E. Leed, "Canonical body knowledge, perceptuo-motor coordination, and tactile localization," Ph.D. dissertation, Tulane University, New Orleans, Louisiana, USA, 2014.

[14] E. Somogyi, L. Jacquey, T. Heed, M. Hoffmann, J. J. Lockman, L. Granjon, J. Fagard, and J. K. O'Regan, "Which limb is it? Responses to vibrotactile stimulation in early infancy," *British Journal of Developmental Psychology*, [in revision].

[15] D. Corbetta, S. L. Thurman, R. F. Wiener, Y. Guan, and J. L. Williams, "Mapping the feel of the arm with the sight of the object: on the embodied origins of infant reaching," *Frontiers in psychology*, vol. 5, p. 576, 2014.

[16] J. K. O'Regan and A. Noe, "A sensorimotor account of vision and visual consciousness," *Behavioral and Brain Sciences*, vol. 24, pp. 939–1031, 2001.

[17] E. Oztop, N. S. Bradley, and M. A. Arbib, "Infant grasp learning: a computational model," *Experimental brain research*, vol. 158, no. 4, pp. 480–503, 2004.

[18] J. Gottlieb, M. Lopes, and P.-Y. Oudeyer, "Motivated cognition: Neural and computational mechanisms of curiosity, attention, and intrinsic motivation," in *Recent Developments in Neuroscience Research on Human Motivation*. Emerald Group Publishing Limited, 2016, pp. 149–172.

[19] N. Dehorter, L. Vinay, C. Hammond, and Y. Ben-Ari, "Timing of developmental sequences in different brain structures: physiological and pathological implications," *European Journal of Neuroscience*, vol. 35, no. 12, pp. 1846–1856, 2012.

[20] O. Herbort and M. V. Butz, "Too good to be true? ideomotor theory from a computational perspective," *Frontiers in psychology*, vol. 3, p. 494, 2012.

[21] H. Sakata, Y. Takaoka, A. Kawarasaki, and H. Shibutani, "Somatosensory properties of neurons in the superior parietal cortex (area 5) of the rhesus monkey," *Brain research*, vol. 64, pp. 85–102, 1973.

[22] M. S. Graziano, C. S. Taylor, T. Moore, and D. F. Cooke, "The cortical control of movement revisited," *Neuron*, vol. 36, no. 3, pp. 349–362, 2002.

[23] T. Heed, V. N. Buchholz, A. K. Engel, and B. Röder, "Tactile remapping: from coordinate transformation to integration in sensorimotor processing," *Trends in cognitive sciences*, vol. 19, no. 5, pp. 251–258, 2015.

[24] S. Badde and T. Heed, "Towards explaining spatial touch perception: weighted integration of multiple location codes," *Cognitive neuropsychology*, vol. 33, no. 1-2, pp. 26–47, 2016.

[25] X. Pitkow and D. E. Angelaki, "Inference in the brain: Statistics flowing in redundant population codes," *Neuron*, vol. 94, no. 5, pp. 943–953, 2017.

[26] M. Hoffmann, H. Marques, A. Hernandez Arieta, H. Sumioka, M. Lungarella, and R. Pfeifer, "Body schema in robotics: A review," *Autonomous Mental Development, IEEE Transactions on*, vol. 2, no. 4, pp. 304–324, Dec 2010.

[27] G. Schillaci, V. V. Hafner, and B. Lara, "Exploration behaviors, body representations, and simulation processes for the development of cognition in artificial agents," *Frontiers in Robotics and AI*, vol. 3, p. 39, 2016.

[28] G. Metta, L. Natale, F. Nori, G. Sandini, D. Vernon, L. Fadiga, C. von Hofsten, K. Rosander, M. Lopes, J. Santos-Victor, A. Bernardino, and L. Montesano, "The iCub humanoid robot: An open-systems platform for research in cognitive development," *Neural Networks*, vol. 23, no. 8-9, pp. 1125–1134, 2010.

[29] A. Roncone, M. Hoffmann, U. Pattacini, and G. Metta, "Automatic kinematic chain calibration using artificial skin: self-touch in the iCub humanoid robot," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, 2014, pp. 2305–2312.

[30] M. Longo, E. Azanon, and P. Haggard, "More than skin deep: Body representation beyond primary somatosensory cortex," *Neuropsychologia*, vol. 48, pp. 655–668, 2010.

[31] M. Hoffmann, Z. Straka, I. Farkas, M. Vavrecka, and G. Metta, "Robotic homunculus: Learning of artificial skin representation in a humanoid robot motivated by primary somatosensory cortex," *IEEE Transactions on Cognitive and Developmental Systems*, 2017.

[32] M. Hoffmann and N. Bednarova, "The encoding of proprioceptive inputs in the brain: knowns and unknowns from a robotic perspective," in *Kognice a umely zivot XVI [Cognition and Artificial Life XVI]*, M. Vavrecka, O. Becev, M. Hoffmann, and K. Stepanova, Eds., 2016, pp. 55–66.

[33] S. S. Kim, M. Gomez-Ramirez, P. H. Thakur, and S. S. Hsiao, "Multimodal interactions between proprioceptive and cutaneous signals in primary somatosensory cortex," *Neuron*, vol. 86, no. 2, pp. 555–566, 2015.

# Appendix E

# Goal-directed tactile exploration for body model learning through self-touch on a humanoid robot

Gama, F., Shcherban, M., Rolf, M., and Hoffmann, M. (2021). Goal-directed tactile exploration for body model learning through self-touch on a humanoid robot. *IEEE Transactions on Cognitive and Developmental Systems*. [early access]

Open Access. DOI: `https://doi.org/10.1109/TCDS.2021.3104881`
Youtube video: `https://youtu.be/dnJaffBHf1c`

Author contributions: The contribution of M. Hoffmann was 20%.

# Goal-Directed Tactile Exploration for Body Model Learning Through Self-Touch on a Humanoid Robot

Filipe Gama, Maksym Shcherban, Matthias Rolf, and Matej Hoffmann, *Member, IEEE*

***Abstract*—An early integration of tactile sensing into motor coordination is the norm in animals, but still a challenge for robots. Tactile exploration through touches on the body gives rise to first body models and bootstraps further development such as reaching competence. Reaching to one's own body requires connections of the tactile and motor space only. Still, the problems of high dimensionality and motor redundancy persist. Through an embodied computational model for the learning of self-touch on a simulated humanoid robot with artificial sensitive skin, we demonstrate that this task can be achieved 1) effectively and 2) efficiently at scale by employing the computational frameworks for the learning of internal models for reaching: intrinsic motivation and goal babbling. We relate our results to infant studies on spontaneous body exploration as well as reaching to vibrotactile targets on the body. We analyze the reaching configurations of one infant followed weekly between 4 and 18 months of age and derive further requirements for the computational model: accounting for 3) continuous rather than sporadic touch and 4) consistent redundancy resolution. Results show the general success of the learning models in the touch domain, but also point out limitations in achieving fully continuous touch.**

***Index Terms*—Body exploration, body schema, goal babbling, intrinsic motivation, reaching development, self-touch.**

## I. Introduction

While recent decades have seen vast progress in robot tactile sensing [1], [2], touch is still rarely used as a primary sense. Research largely focuses on tactile sensing in robot hands for manipulation [3], which is overseen and guided by vision. Systems with the ambition of comprehensive touch as a primary sense are still the exception; if present, touch is an add-on to existing sensing systems (e.g., in the iCub humanoid robot [4]). In stark contrast, touch is the first sense to emerge in the biological fetus [5] and provides a crucial scaffolding by which later motor and cognitive development are grounded [6]. For touch to be used truly as a primary sense, it is necessary to mutually ground and coordinate tactile sensing, vision, and motor control, which poses an enormous challenge to the development of robotic systems. Self-touch may play a crucial role for the development of these skills in infants [7], and indeed seems like a uniquely well-suited calibration scenario: all three modalities are used and create contingent [8] stimuli, which are much less dependent on uncontrollable external factors than when touching external objects. Generating self-touch requires coordinated motor action, which can be challenging for systems with many Degrees of Freedom (DoF) and sparse sensor arrangement. This is true in particular for robots as their tactile sensing is still rather rudimentary and self-contact does not occur naturally like in infants.

The overarching hypothesis of this work is that the similarities and shared challenges between self-touch and reaching for objects external to the body demand for a shared conceptual, technological, and developmental framework. This leads us to the *technological hypothesis* that learning coordinated self-touch can be achieved with models initially proposed for external reaching. Learning to coordinate one's body by means of inverse models [9] that suggest motor actions for desired outcomes is a well-understood concept for external reaching. We seek to transfer knowledge from this well-understood domain to the much less understood touch domain, and transfer specific concepts such as goal-directed and structured exploration in order to allow mastery of highly articulated motor systems. Learning inverse models through exploration of random movements—often dubbed *body babbling* [10] or *motor babbling*—has been employed in different models (e.g., the "endogenous random generator" in [11]). However, faced with the dimensionality of the motor and sensory spaces, trying out all possible combinations of motor commands and observing their consequences is hugely inefficient. For example, most motor commands generate movements that do not result in any contact with the body and hence do not generate useful experience to learn the motor–tactile contingencies. Therefore, we employ two key ideas that help the agent to channel the exploration in the right direction. First, the agent should monitor its learning efficiency—the gain in its knowledge or competence to achieve specific goals—and

Manuscript received October 1, 2020; revised March 31, 2021 and June 23, 2021; accepted July 27, 2021. This work was supported by the Czech Science Foundation (GA CR), Project EXPRO under Grant 20-24186X. *(Corresponding author: Matej Hoffmann.)*

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Tulane University Social-Behavioral IRB under Application No. 153903 and granted by the Ethics board of the Institute of Psychology, Academy of Sciences of the Czech Republic under Application No. PSU-300/Pha/2020.

Filipe Gama, Maksym Shcherban, and Matej Hoffmann are with the Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, 16627 Prague, Czech Republic (e-mail: filipe.gama@fel.cvut.cz; matej.hoffmann@fel.cvut.cz).

Matthias Rolf is with the School of Engineering, Computing and Mathematics, Oxford Brookes University, Oxford OX3 0BP, U.K. (e-mail: mrolf@brookes.ac.uk).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCDS.2021.3104881.

Digital Object Identifier 10.1109/TCDS.2021.3104881

focus the exploration on regions of the search space that are currently most promising. This is exemplified by the computational frameworks dealing with intrinsic motivation (or artificial curiosity) [12]–[15]. Second, the agent should focus the exploration on the goal space (hence "goal babbling") rather than the motor space [14], [16]. The goal space—the skin on the body in our case—may be lower dimensional and it is where the "interest" of the agent lies. The key challenge that self-touch adds to the learning problem, compared to reaching, is that most motor actions do not result in any observable self-touch outcome in any region of interest, and that sensible outcomes only lie on a lower dimensional manifold (the skin) in space [17]. In contrast, typical reaching setups [14], [16] involve sensible visual outcomes at least for the majority of possible actions spanning the entire volume of space, which provides much denser and richer feedback.

Complementary to the technological hypothesis, we promote the *developmental hypothesis* that a continuity of mechanisms exists between self-touch and reaching in infant development, rather than distinct mechanisms in each respective stage of development [7]. This aligns with the idea of goal babbling, which itself is a continuity model [18] of exploration and exploitation.

### *A. Related Work*

A key requirement for a model to learn self-touch is *efficiency*: the learner has to cope with limited learning time and resources and learns to coordinate in a potentially high dimensional motor space. Efficient learning of general sensorimotor skills has been studied extensively. Our focus is "mechanisms that drive a learning agent to perform different activities for their own sake, without requiring any external reward" [14]. This phenomenon has been articulated in psychology as intrinsic versus extrinsic motivation—[19] provides an overview. Oudeyer and Kaplan [13] strived to clarify the terms of internal/intrinsic and external/extrinsic rewards and present a computational perspective as well as the relationship to other computational frameworks such as reinforcement learning. As briefly outlined above, there are two key aspects of efficient exploration: 1) monitoring learning progress and 2) focusing on the "goal space." The former has been addressed by a number of frameworks that can be classified as *knowledge-based* [13]. The latter aspect has been addressed by the goal babbling approach of Rolf *et al.* [16] or by other *competence-based* approaches, in which the agent self-generates goals that it tries to accomplish. The idea is best illustrated on the example of learning to reach, or learning inverse kinematics. The motor system is known for its redundancy: there are multiple ways of reaching to a specific point in space. Knowledge-based approaches that monitor learning progress but are confined to the motor space (e.g., [20]) will discover multiple solutions to the same goal, which can often be considered inefficient. Moreover, the space of solutions in the joint space (motor space) is not convex: averaging between them will often result in wrong configurations. Rolf *et al.* [16] analyzed this and developed a solution, goal babbling, that deals with this problem: by exploring in the goal space, the agent is not "motivated" to look for alternative solutions. Furthermore, following continuous paths through the goal space allows to circumvent the issue of nonconvex solutions [16]. This architecture has been also used to model the U-shaped curve typical of infant development [21]. Technologically, it has enabled to address the control of advanced biomimetic robots such as an elephant-trunk-inspired robot [22]. Baranes and Oudeyer extended their robust intelligent adaptive curiosity (R-IAC) architecture [20] to self-adaptive goal generation R-IAC (SAGG-RIAC) [14]—a competence-based strategy—that also handles learning inverse kinematics in redundant manipulators. Our work is employing the computational framework of [14], as embedded in the *Explauto* library [23]. The algorithms for both modeling and exploration embedded in *Explauto* heavily rely on motor actions being discontinuous in time, whereas path-based approaches [16] rely on continuity and exploit it. Very recent work in [24] has, however, seen a first integration of path-based goal exploration [16] and competence-based intrinsic motivation [14].

Learning to discover the surface of the body—a 2-D skin surface embedded in the 3-D world and moving together with the body parts—is similar to the problem of learning inverse kinematics that is a typical showcase for many of the intrinsic motivation frameworks (e.g., [14] and [16]). The motor space or joint space is identical; the goal space, or *observation space*, is different: for learning inverse kinematics, these are the 3-D Cartesian coordinates of the end effector (e.g., the infant hand). For the body space, either skin activation or spatial coordinates are candidate representations, which will be explained in detail in Section III. The key difference to reaching in general is that during reaching to one's own body, feedback is not available for many postures. Some, or most (depending on morphology), motor actions will not result in a significant self-touch, in which case there may not be a possible learning step for a model. Whether existing frameworks (e.g., [14] and [16]) can be *effective* (i.e., functional) under these specific circumstances has not been shown so far.

The work of Mori and Kuniyoshi [25] and Yamada *et al.* [26] on the fetus simulator is complementary to this work, addressing prenatal development and focusing on a lower level: first tactile–motor interactions are emerging from the musculoskeletal body model coupled to spinal and simple subcortical or cortical circuitry. In comparison, the present study focuses on how guided exploration on a higher level of abstraction can give rise to efficient body exploration. More importantly, we are looking for goal-directed coordination that will facilitate comparison with infant data from experiments where dynamically chosen locations were designated as touch worthy. The model in [25] and [26] does not comprise any inverse models and therefore does not allow for goal-directed movement, but scales by only looking for broad correlations on statically defined interest patterns.

The work most related to ours is that of Mannella *et al.* [27] who specifically target the body (skin surface) as the exploration target. The biggest experimental difference to the present study is the scale of the task. While we are looking at a humanoid scale motor problem with 2-D skin, Mannella *et al.* only investigated a very simple simulation. It consisted of two arms in 2-D with three DoF each, and a "skin" emulated using 30 Gaussian receptive fields in a 1-D topology. Their architecture is rather more complex compared to ours, consisting of a Goal generator, Goal selector, Motor controller, and Predictor. The motor controller is also highly complex, composed of a dynamic-reservoir recurrent neural network, a random generator, and associative memory. The "skin receptors" are phasic, as they respond to changes rather than sustained values. These changes are then relayed into a self-organizing neural map (SOM) that "clusters" them. Coordination is achieved by learning motion trajectories through reinforcement. The inner number of variables, or dimensions, is therefore rather high, which makes scaling to humanoid complexity challenging. Compared to this, our architecture is much simpler *and chosen specifically for efficiency* by focusing on direct inverse models that can estimate the necessary posture directly, rather than having to learn to generate entire trajectories. The motor space consists simply of the robot joint space. That is, only the final configurations/postures matter—motor overlaps with proprioceptive—and the actual movement production is sidestepped. Additional discussion and pointers to related work on this issue—learning to reach "synchronously" by matching points in space versus "asynchronously," taking into account movement trajectories, is presented in [28].

This work is a direct extension of our previous work [29], adding in particular: 1) pilot analysis of infant reaching kinematics for targets on the body (Section II); 2) path-based goal babbling (after [22]); 3) improvements of the experimental setup; and 4) new experiments and analyses.

### *B. Structure and Outline*

Section II reviews relevant behavioral studies on infants and presents our own pilot study on reaching configurations used by an infant to reach for targets on the body. We distill two further key requirements to a computational model for the development of self-touch: 1) accounting for not just singular touches, but continuous touching motion and 2) describing a consistent and repeatable choice of redundancy resolution. Section III presents the robot simulator and the exploration framework. Experimental results (Section IV) show for the first time the effectiveness of existing intrinsic motivation and goal babbling frameworks in the touch domain, as well as their efficiency. We summarize the key findings from the experiments in Section V and discuss their implications and future work in Section VI.

## II. Infant Body Exploration and Reaching for Targets on the Body

Fetuses initially perform local movements directed to areas of the body most sensitive to touch: the face, but also soles of feet [30, pp. 113–114]. Later, from 26 to 28 weeks of gestational age, they also use the back of the hands and touch other body areas like thighs, legs, and knees [30, pp. 29–30]. In addition, from 19 weeks, fetuses anticipate hand-to-mouth movements [31] (the mouth opens prior to contact) and from 22 weeks, the movements seem to show the recognizable form of intentional actions, with kinematic patterns that depend on the goal of the action (toward the mouth versus toward the eyes) [32].

Hand–mouth coordination continues to develop after birth [33]. Specifically related to body exploration, Rochat [34] writes: "By 2–3 months, infants engage in exploration of their own body as it moves and acts in the environment. They babble and touch their own body, attracted and actively involved in investigating the rich intermodal redundancies, temporal contingencies, and spatial congruence of self-perception." Thomas *et al.* [35], biweekly recording resting alert infants from birth to six months of age, showed that infants frequently touch their bodies, with a rostrocaudal progression as they grow older: head and trunk contacts are more frequent in the beginning, followed by more caudal body locations including hips, then legs, and eventually the feet. DiMercurio *et al.* [36], following infants from 3 to 9 weeks after birth, found no consistent differences in the rate of touch between the head and the trunk. In summary, infants acquire ample experience of touching their body, which allows for the learning of the first tactile-proprioceptive-motor models of the body. The ability to learn from this experience goes hand in hand with dynamic neural development in this period [37]; see [7] for a review focusing specifically on self-touch. Yet, the behavioral organization of such early tactile exploration not understood. Are the touches on the body spontaneous or systematic? If there is a particular structure—which seems to be the case [35], [36]—what drives this developmental progression? Piaget [38] theorized that in newborns, action and perception as well as the "spaces" of individual sensory modalities are separated (cf. [39] for evidence that visual and motor modalities are connected early after birth). Until the connections are established, infants explore their environment (and their body) randomly. Piaget [38] also proposed a pivotal role of repeated movements—*primary circular reactions* directed to learn properties of the body and *secondary circular reactions* driven by the interest on the effects they produce in the environment. A computational account combining rhythmic and discrete movements and reinforcement learning is presented in [40].

Behavioral studies investigating infant spontaneous behaviors with a specific focus on touch to the body [35], [36] provide data that inform our modeling. First, in the first weeks after birth, contacts with the rostral areas of the body (head and trunk) are dominant. Second, contacts are typically made with the ipsilateral hand. DiMercurio *et al.* [36] also identified *complex touches* (as infants moved their hand while remaining in contact with their body), performed a network analysis of contact sequences, and identified points of centrality on the body. However, to discriminate spontaneous contacts from systematic (intrinsically motivated) exploration remains a challenge. The observation of frequent "complex touches" raises an important requirement for a computational model—it has to explain the *continuous* coordination of the task across isolated touch points. Purely conceptually, path-based exploration approaches [18] seem to fit the requirement of exploration "along the skin surface" more naturally than sporadically sampling ones [14], where every movement is started anew from some canonical ("home") posture. However, sporadic exploration at some point also has to move between points, even though that is typically abstracted away and only end points are considered. Which approach creates experimental data closer to infant observations is not clear yet, partially pending further quantitative analysis of infant data.

A counterpart to recordings of spontaneous infant behavior is provided by testing how they can reach to targets on their body. Lockman and colleagues performed a series of studies [41]–[45] in which vibrotactile targets (“buzzers”) were attached to infants’ body parts and their ability and their way of reaching for the targets were analyzed. Targets above the mouth and on the chin were successfully contacted already from 2 months of age [42], followed by trunk area, legs, hands, other areas on the face (forehead and ears), and elbows (around nine months) (whole body—pilot study [43]; upper body [44]). For targets on hands and arms, the arm with the buzzer and the contralateral arm reaching for the target often moved simultaneously—the arm with the target actively facilitating the removal [41].

One aspect remained unexplored so far: how infants manage motor redundancy in this task. Do they use the same arm configuration to reach for specific targets on the body or do they have alternatives at their disposal? If the latter is true, what does the choice of a solution depend on? Switching solutions may depend on the initial/current posture—what is known as motor hysteresis [46]. Finally, what is the developmental progression of this phenomenon? This kind of information is important for the modeling work, as it constraints the inverse model—a mapping from body space to joint space—representation. To this end, we performed an analysis of reaching configurations of one infant tested weekly with buzzers on her body, between 4 and 18 months (experimental protocol described in [44] in detail; summary of success in buzzer removal for this infant reported in [43]; study approved by Tulane University Social-Behavioral IRB). From video recordings of the experiments, snapshots of final postures—around the first contact with the target—were taken. In this article, reaching to the face and trunk will be studied. Hence, we show the final postures the infant assumed to reach for the right forehead (Fig. 1, top) and the left trunk (Fig. 1, bottom). Interestingly, despite the fact that the overall posture of the infant differed significantly (supine, seated in a seat/on the lap, on the tummy, and freely sitting) and that the snapshots span more than one year of the infant’s life, the final reaching configurations are largely similar. Thus, the constraints on the inverse model can be relaxed. Although it seems clear that, at least eventually, alternative reaching configurations will be available [46], they are not frequently spontaneously recruited. This observation motivates the fourth key requirement: the model needs to describe a *consistent* resolution of redundancy, rather than selecting different postures in every attempt or across trials. This is consistent with the notion of a direct inverse model which stores exactly one solution to reach for any goal. However, consistent behavior may be distally observable also for different internal organizations of the motor control skill.

## III. Embodied Computational Model—Materials and Methods

This section provides an overview of the robot simulator and the exploration framework.

### A. Nao Humanoid Robot With Artificial Skin

The experimental platform of this study is a Nao humanoid robot. A modified version, uniquely equipped with artificial sensitive skin (Fig. 2), is available at our institute. The electronic skin covers the robot’s wrists, torso, and head (Fig. 2, right). Every triangular module hosts ten pressure-sensitive elements, or “taxels” [47]. The physical robot is currently not employed in the results shown here—long exploration experiments would be damaging to the hardware—but the simulated version closely mimics it to allow the transition to the real robot in the future.

Gazebo 9 simulation environment was used. A variant of the publicly available *naov40* URDF model was additionally fitted with tactile/pressure sensors (“skin”), mimicking our physical Nao robot (Fig. 2). In addition, a cylindrical “pen” tool with a spherical endpoint was attached to the robot wrist to act as a finger and facilitate localized touch. The model has 250 and 240 tactile sensors for the torso and the head, respectively (see Fig. 3). Compared to [29], additional modifications were performed to the simulator that included simplification of the meshes composing the model, buffering the contact events, and disabling collisions for other parts of the arm than the finger. Overall, these changes resulted in faster execution and more stable results. The code of our `nao-gazebo-skin` plugin is available at [48]. Videos from our experiments are available at https://youtu.be/dnJaffBHf1c.

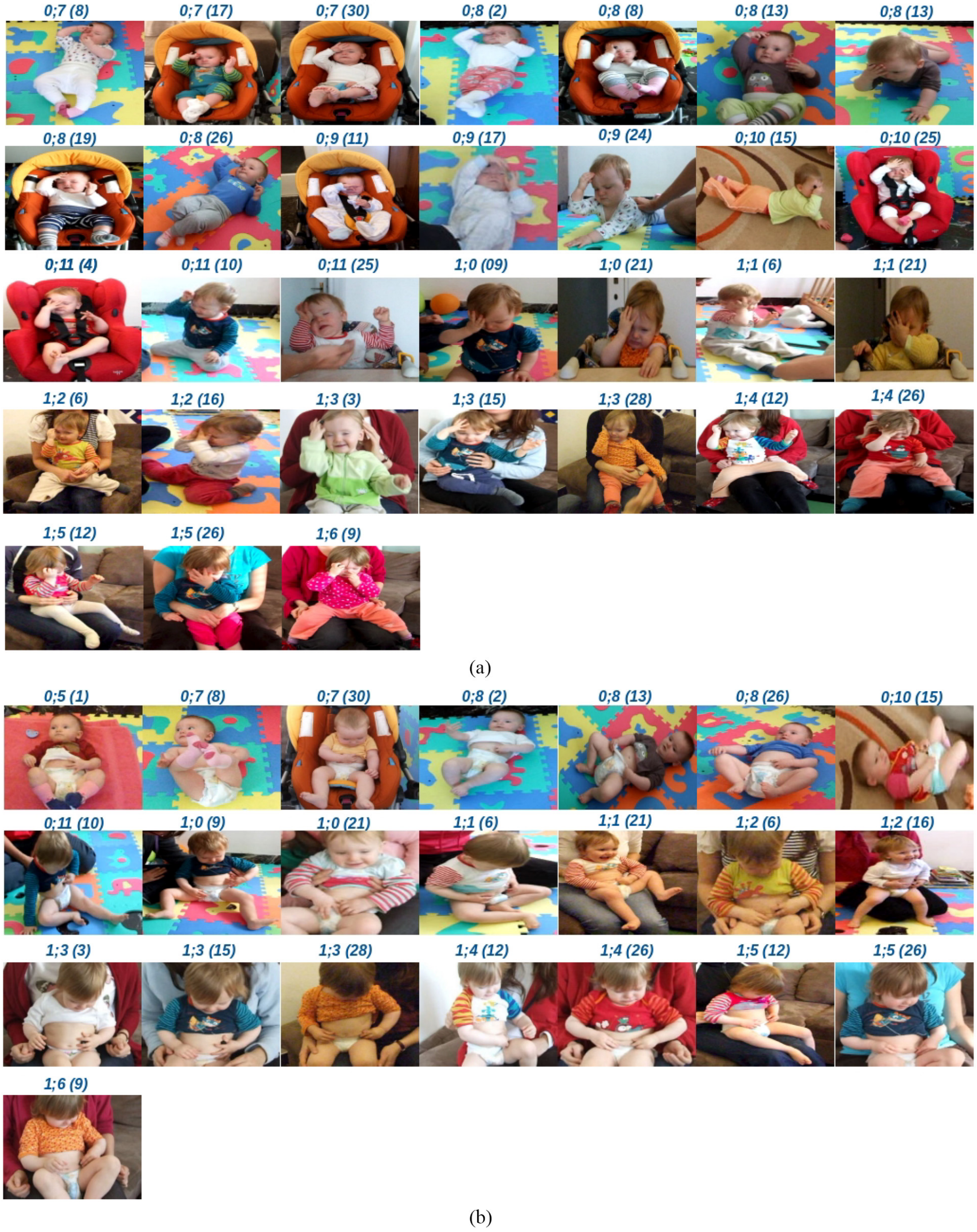
### B. Action and Observation Spaces

The *action space* $Q$ consists of the robot joint angles. An action $q \in Q$ generates an outcome $x \in X$ in the *observation space* $X$. Only the upper body of the Nao robot, which hosts the artificial skin, is used. The robot uses its right arm to touch either the torso or the head. Its *action space* is the robot’s joint space, with five DoF per arm and two DoF on the neck. To touch the torso, only the arm is used, hence $Q \subseteq \mathbb{R}^5$; to touch the head, the neck joints may also contribute: $Q \subseteq \mathbb{R}^7$. The effect of these two joints is specifically studied by comparing with a configuration without the head joints (*nohj*). Position control in every joint is used to command the simulated robot to a desired joint configuration.

The *observation space* is the robot skin activation generated when the robot contacts its torso or face with its arm. This is a discrete space of individual taxels and their activation (binary: activated or not). For the exploration methods considered here, a distance *metric* on this space is needed. A parallel planar projection (Fig. 4) is used (differently from [29] where a cylindrical projection was used for “high-resolution” skin). Thus, for the torso and head skin, $X \subseteq \mathbb{R}^2$.

### C. Touch versus Motor and Task Errors

Not all actions or postures in this scenario result in a measurable touch outcome. Fig. 5 illustrates various cases for the



(a)

(b)

Fig. 1. Infant reaching to the body. Screenshots of final postures while reaching for a buzzer on: (a) right forehead and (b) left trunk. One infant followed from 4 to 18 months. Only screenshots from sessions where she successfully contacted the target are shown. Age is reported in the notation years; months (days).

simplified scenario of a 2 DoF planar arm, but for which the action space can be fully plotted. A touch signal is generated if the tip of the arm gets in contact with the surface of the object. The observation space of 1-D touches is color coded for different touch coordinates $x$ together with actions that achieve precisely that touch outcome (shown both as postures

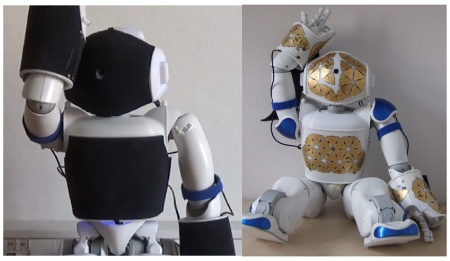


Fig. 2. (Left) Nao robot with skin performing self-touch. (Right) Robot with exposed artificial skin.

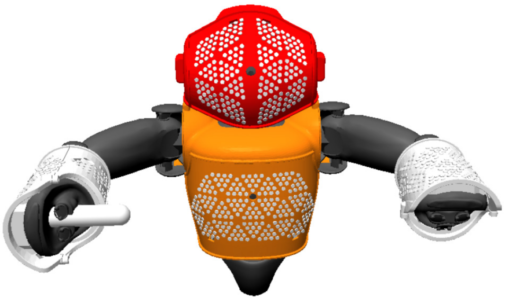
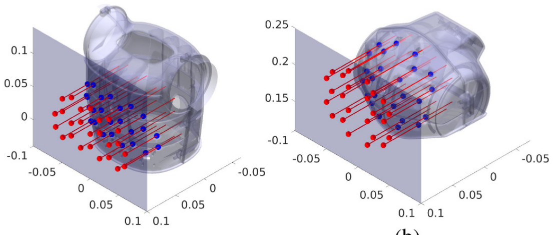


Fig. 3. Nao robot model in Gazebo with tactile sensors.



(a)

(b)

Fig. 4. Parallel planar projection of taxels' coordinates for the (a) torso and (b) head.

in the physical space, and action coordinates in $Q$; note that the physical space is 2-D while the observation space is only unidimensional—contour of the gray object). Some postures, such as (1), do not generate a touch. Others, (2), would penetrate the body and are impossible (the entire grayed out area in the action space). Some postures generate physical contact, (3), but may not be registered as a touch sensation if performed with an area without sensors, such as the elbow.

Fig. 5, bottom, shows different errors and deviation that can occur when performing a goal-directed touching motion. The robot in this example could start from a neutral reset posture. Trying to reach for some touch goal $x^*$, the inverse model suggests a posture $q^*$. The underlying motor controller is tasked with moving the robot to $q^*$, which is physically unreachable in this example. Instead, the arm at some point collides with the skin and is potentially dragged along it while the motor controller is still trying to reach $q^*$. The robot eventually ends up in a different posture $q$, with a touch coordinate $x$ (but may generate a nontouch event in other cases). When attempting an impossible action $q^*$, a motor discrepancy—difference between the desired joint configuration $q^*$ and actual configuration at the end of the movement $q$—is inevitable. A different kind of error is to miss the reaching target $x^*$ by $x$. This particular motion is an example of a single *sporadic* touch. Complex, or continuous, touches would describe a motion between several different goals $x^*$ while maintaining touch with the skin. During such complex motions, the motor discrepancies can be subject to hysteresis as the effector is dragged along the skin.

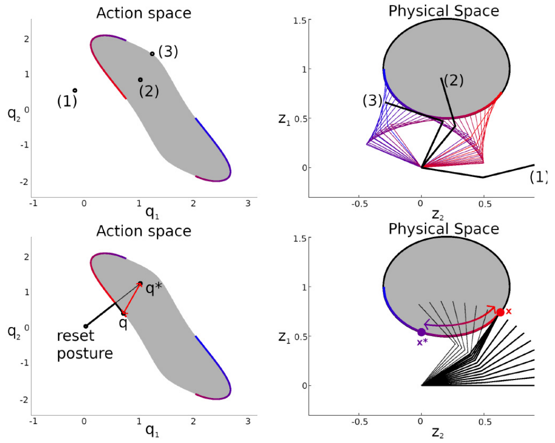


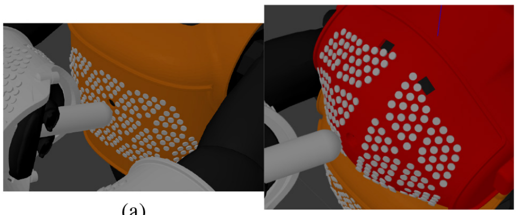
Fig. 5. Exemplary relation between the action space of a 2 DoF arm, and its potential touch on the 1-D boundary (observation space) of a body. Top: Examples of different types of touch (color coded) and nontouch events (numbered)—a nontouching posture (1), an impossible posture (2), and a posture that generates a contact with the object that does not register as touch (3). Bottom: Exemplary reaching attempt to a rest or reset posture with both a reaching error in the observation space, and a motor discrepancy.

### D. Home Posture

The default "home" posture of the model in Gazebo that is used to start reaching movements is similar to the one shown in Fig. 3, with the difference that the wrist is rotated such that the "finger" points downward. For exploration methods relying on separate movements to collect every action-observation pair—referred to as *sporadic touch* here—this posture is of little importance, provided that this posture does not impede reaching to the goal space (torso or head) due to self-collisions. However, for path-based continuous approaches (Section III-F), the home posture is an important part of the exploration process and it is beneficial if it lies in the goal space. To this end and in order not to introduce biases or asymmetries, home postures with the hand touching the central part of the torso/head were chosen (see Fig. 6). This home posture is used both for path-based continuous goal babbling (PBCGB) and as the base posture to bootstrap the Explauto methods.

### E. Explauto Autonomous Exploration Library

*Explauto* (https://github.com/flowersteam/explauto, [23]) is a framework for implementation and benchmarking of


(a)

(b)

Fig. 6. Home posture providing a touch feedback for the (a) torso and (b) head.

sensorimotor learning algorithms, with a specific focus on intrinsic motivation—monitoring learning progress in motor or sensory (goal) spaces. For a detailed and formal explanation, the reader is kindly referred to [14], [23], [49], and [50]. During exploration, a database is constructed with every entry being a tuple: $(q, x)$. We use the *nearest neighbor* model from Explauto to represent both the forward and inverse model (together, the sensorimotor model in the terminology of Explauto). This is an example of *lazy learning*: training data is processed only when a query is asked. Our focus is on inverse models: learning how to reach for particular locations on the skin ($\sim$ inverse kinematics). Given an observation $x^*$, the inverse model will return the joint configuration $q^*$ that corresponds to the observation stored in the database that is closest to $x^*$. Forward models are needed only for the discretized motor babbling (DMB) exploration strategy (see below) that uses forward predictions to gauge the learning progress.

To choose where to focus exploration, an *interest model* is needed, sampling from the *interest space*, which can be either the action space $Q$ (motor babbling strategies) or the observation space $X$ (goal babbling strategies). The exploration strategies from Explauto used in this work are as follows.

*1) Random Motor Babbling:* A motor configuration $q^* \in Q$ is sampled uniformly from the action space, and then executed, generating an observation $x \in X$.

*2) Random Goal Babbling:* A goal $x^*$ is sampled uniformly from the observation space, and the inverse model is used to find an action $q^*$ best matching the goal, with added exploration noise.

*3) Discretized Motor Babbling (or Motor Babbling With Intrinsic Motivation):* The interest space—the action space—is discretized into $c = m \times n$ cells (regions). We use $32 \times 32$ cells. The algorithm randomly selects one of the cells with a probability proportional to the current state of an *interest value* $I$ that each cell possesses. A motor command $q_c^*$ within that cell is generated. The forward model is used to make a prediction of the observation, $x_c^*$. The motor command is executed, resulting in the real observation $x$ and allowing for the calculation of an error (expected versus real). The interest value of a cell is high when *competence* $C = ||x_c^* - x||$, rapidly increases or declines. The *local competence progress* is formally defined in [49].

*4) Discretized Goal Babbling:* This method is analogous to DMB but with the interest space being the observation space (the skin) and the competence progress gauged on the performance of the inverse model. The goal generation randomly selects one of the cells, proportionally to their interest value $I$. Then, a goal $x_c^*$ within that cell is uniformly generated. The robot attempts to reach for $x_c^*$ using the inverse model (with exploration noise).

All experiments using *Explauto* were run for 1000 iterations. Random goal babbling (RGB), DMB, and discretized goal babbling (DGB) require a bootstrapping phase, during which a few touches have to be generated. This phase is counted toward the 1000 iterations limit. To this end, random motor babbling (RMB) with constrained joints range is used by taking the *home posture* as a base and adding random exploration noise that fits within the constrained range until ten touches are observed. While a single touch was enough to bootstrap these methods, the use of additional bootstrapping touches led to more consistent results between trials of the same experiment (with small improvements from 5 to 10 touches and no effects with additional ones).

### *F. Path-Based Continuous Goal Babbling*

In addition to *Explauto* and related SAGG-RIAC methods, we employ the PBCGB method originally developed in [16] with modifications previously employed on the biomimetic elephant trunk robot [22]. While *Explauto* explores by means of sporadic random motion and discontinuous goal selection, [22] samples motor action along continuous paths. A goal $x_E^*$ is chosen randomly from the overall set of goals as the end point of a motion. From the previous end point $x_S^*$, a straight path of subgoals $x^*(t)$ is then sampled toward $x_E^*$. Each subgoal $x^*(t)$ along the path is attempted to be reached by querying the inverse model for a motor command $q_t^*$. A small random motor perturbation $\epsilon(t)$ is added to the command to encourage exploration. In order to ensure continuity, this perturbation is not chosen independently in every time step, but generated by an autoregressive random process similar to a Brownian motion that can explore over a wide range of values by accumulating small random steps (see [16], [22] for details). The perturbed motor action $q^*(t) + \epsilon(t)$ is then executed on the robot in order to try and reach for $x^*(t)$ along the continuous goal path. The resulting observation $x(t)$ is then used together with the motor command $q^*(t) + \epsilon(t)$ to train the inverse model. After attempting to reach the end point and each subgoal in its path, either another end point is selected, or, with a fixed probability, the *home posture* is designated as the next end point.

This is illustrated in Fig. 7: we start from the home posture (red dot in the center, 29), generate the first end point (red dot 7), with subgoals between it and the home posture (blue dots from 1 to 6). The model attempts to reach for 1, then 2, until 7. After the attempt to reach for 7, a new end goal is generated (red dot 15), with its subgoals. This happens again to reach for 22, at which point the home posture was chosen instead of another end goal. When the algorithm goes back to the home posture, it uses the difference between the current position and the home posture to reach it in steps—here, 7 steps from 22 to 29—but without generating subgoals. This

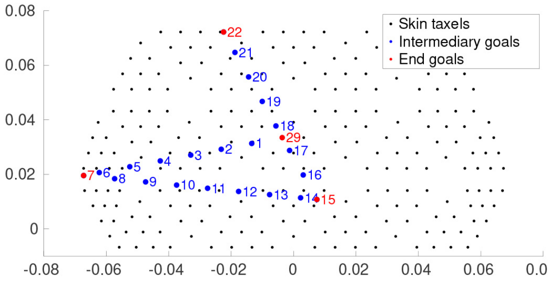


Fig. 7. Example of exploration paths and goals generated by PBCGB. End points in red, subgoals in blue, and taxels in black.

ensures that the robot is back on the home posture to start a new exploration path no matter where it was, as there is no guarantee that the goals were successfully reached.

An advantage of this continuous strategy is that the way redundancy is resolved by the eventual inverse model is typically very smooth and does not involve unnecessary jumps. It can moreover be controlled by choosing a home posture, from which exploration starts and to which it repeatedly returns. If, for example, the home posture is chosen in an "elbow up" configuration, the path-based approach will develop an inverse model that stays close to it and moves continuously around it to reach goals. SAGG-RIAC [14] cannot achieve this continuity because its sporadic random actions will sample elbow up and "elbow down" postures equally, and the Nearest Neighbor model has no means to reconcile them smoothly. Hence, the inverse model would suggest repeatedly switching between both kinds of redundancy resolutions while trying to generate a continuous motion.

An important implication of focusing on a single, consistent way of resolving redundancy is that this strategy has been shown to work in extremely high-dimensional motor spaces, such as 50 [16] or 100 [51] motor dimensions. The main drawback is that the creation of continuous paths and explorative perturbations requires more parameters (e.g., step widths and amplitudes) than the sporadic exploration of SAGG-RIAC which only requires to know the outer size of the goal and motor spaces. The path-based approach is therefore more difficult to use "off the shelf." However, the path parameters typically allow for effective experimentation over a wide range of values [16]. The parameters used here were:

1) discretization of the observation space, generating the main end goals (not subgoals): 10 × 10 list of end goals, uniformly spread on the space;
2) probability to go back to the home posture instead of reaching for another end goal: 10%;
3) maximum distance between two successive subgoals, in meters: 0.01;
4) maximum distance joints can move in a single step, in radians: 0.1;
5) motor perturbation: maximum amplitude of added noise that can influence a joint's position, in radians: 0.01.

### G. Learning and Testing Models

In each experiment, the robot uses a given exploration strategy for a certain number of steps: 1000 for *Explauto* methods, and 15 000 for PBCGB. Unlike standard cases in which every iteration of active exploration results in reaching a point in the observation space and allows for calculating an error (target versus actual outcome—like in the case of learning to reach using vision), in our case, the movement does not always result in contacting the skin. In that case, the step is counted toward the maximum number of iterations but does not contribute to learning the sensorimotor (forward or inverse) model.

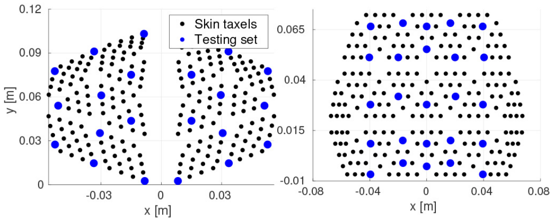


Fig. 8. Testing set. Head (left); Torso (right). All skin taxels in black; the testing set in blue.

At each step, the target joint configuration (motor command sent to the robot) and the actual joint configuration are compared. If the difference for any joint is higher than 10% of its maximum range, the results are discarded and are not used to learn the model. The main reason for such discrepancy is self-collision. Motor commands that would end up "inside the robot" should be avoided. Because of collisions, such commands often make the finger slide on the skin, creating an important difference—sometimes equivalent to the distance between two or three taxels—between the initial contact point and the end point after sliding. This sliding behavior should be avoided as it can influence the results in several ways. For example, an initial contact that does not provide tactile feedback could—if the sliding finger activates a taxel a short time after—be recorded and counted as successfully reaching for the activated taxel.

A subset of taxels, shown in Fig. 8, is tested every 100 iterations during the first 1000 iterations; then every 1000 iterations up to 15 000 in case of PBCGB, whereby testing refers to the robot being asked to reach for each test taxel using the current inverse model. In the same manner as for the exploration phase, if no taxel is contacted, no error can be measured. For all experiments, the environment is set to reset at each iteration during the testing phase, and set back to its previous reset settings before exploration starts again.

## IV. Experiments and Results

We present the results of a series of experiments. The right hand is the robot's effector and reaches either for the torso skin (Section IV-B) or for the face skin (Section IV-C). In the latter case, the action space will be larger as two neck joints are available in addition to the five arm joints. Videos illustrating our experiments are available at https://youtu.be/dnJaffBHf1c.

We will illustrate the results in the following ways: 1) mean reaching error (MRE) every 100 iterations until 1000, then every 1000 iterations (for PBCGB) (e.g., Fig. 12, left); 2) the number of touch occurrences during exploration every 100

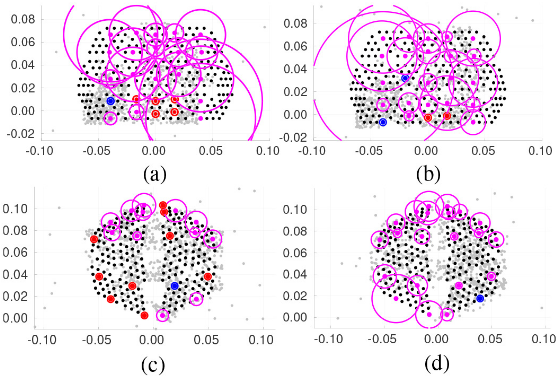


Fig. 9. Right hand reaching for (a) and (b) torso and (c) and (d) head—observation space. Comparing the mean reaching performance over five trials between using the actual joint configuration for learning (left) and the target joint configuration (right). *Goals* generated during the exploration process (gray). Testing after all learning iterations: *Reached taxels* with no error (blue); *Reached with error*—taxels reached (magenta dots) with error magnitude (magenta circles); and *Unreached* taxels—no taxel reached during reaching attempt (red). (a) DGB actual joints. (b) DGB target + actual. (c) DGB actual joints. (d) DGB target + actual.

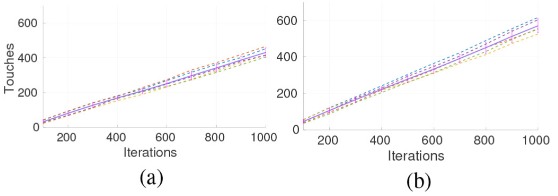


Fig. 10. Right hand reaching for the torso. Touches generated during exploration using the actual joint configuration (left) and the target joint configuration (right). Each dashed line corresponds to a trial. The full line corresponds to the mean, with error bars corresponding to the standard deviation. (a) DGB actual joints. (b) DGB target + actual.

iterations until 1000, then every 1000 iterations (e.g., Fig. 12, right); and 3) the projection of the generated goals with details about the reaching error for each test taxel after 1000 (or 15 000) iterations (e.g., Fig. 9). The results are averaged over ten trials for each exploration strategy; any result specific to one trial or averaged over a different amount of trials will be clearly stated. For projections, the observation space is presented from the point of view of an observer looking at the robot—like in Figs. 2 and 3.

For the experiments on the torso, the *Explauto* exploration methods (Section III-E) and the path-based approach (Section III-F) are complemented by a special version of DGB with no reset, *DGB-nr*, whereby the environment is not reset at each iteration and thus the robot starts from the previous configuration—partially mimicking "complex touches" and the operation of PBCGB.

For the head, DGB, PBCGB, and *DGB-nr* were run. In addition, two more experiments were added: first, a version of DGB, *DGB-nohj* ("no head joints") for which the neck joints were disabled. Second, a version of PBCGB, *PBCGB-incgs* ("increased goal step") with a modified discretization (9 × 9), and maximum distance between two successive subgoals (0.025). The significance of the parameters necessary to organize paths in PBCGB has not so far been exhaustively explored, which gives rise to the expectation that further tuning could enhance its performance. When using the same parameters used for the torso, we observed many subgoals being created in the central area of the head, where there are no taxels. We decided to include both variants here to compare their results.

The difference between RMB, RGB, and DGB in [29] and this work are 1) the projection of the observation space, impacting the calculation of the reaching errors; 2) the discarding of target motor commands that are too different from the actual motor configuration; and 3) changes to the simulator described in Section III-A.

### A. Analysis of Motor Discrepancies

Unlike for reaching in free space, reaching to the body inevitably results in self-collisions. Interesting preliminary results arose related to the comparison between target and actual joint configurations and the discrepancies between them. To avoid the sliding effect (Section III-G) and improve the consistency between what the models learn and what the robot actually does, we tested feeding the models with the actual joint configuration after a command instead of the target command. Over five trials, we observed a significant drop of performance and compared the results with another variant where we used the target motor command by default and replaced it with the actual joint command when the discrepancy for at least one of the joints was 10% higher than its absolute range of possible values. The difference in performance was barely noticeable for the torso, but striking for the head, as shown in Fig. 9, as a significant amount of taxels are unreached when using the actual joints, and reached with small errors when using the target joints.

The difference is not only present in the reaching performance but also, maybe more importantly, in the number of touches generated: for DGB (both head and torso), there is a +50% number of touches between the two cases, on average 400 touches when using the actual joints, and slightly above 600 for the mixed target and actual joints version, as seen in Fig. 10. These results were observed not only for DGB, but also for PBCGB (though only with two trials)—both for the torso and the head. For PBCGB, using mainly the target motor command provided about twice as many touches than using only the actual joints after 15 000 iterations, but the variance between trials was higher.

Thus, commands that try to "penetrate inside the robot" are important to ensure more consistent tactile feedback, as these commands typically exert more force by trying to reach an impossible position, contrary to commands using the actual joint configuration that may end up only brushing against the skin, and not applying enough pressure. To keep the advantage of these commands while reducing the occurrence of the sliding effect and its impact on the results, the solution described in Section III-G was chosen, where the target motor configuration is used but entirely discarded (instead of being replaced

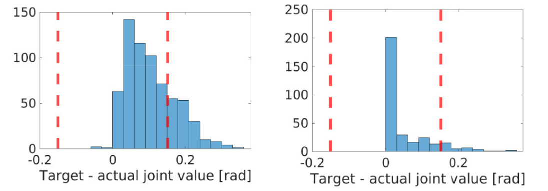


Fig. 11. Joint error through all iterations of the Elbow Roll joint in a single trial from DGB when reaching for the torso. The red vertical lines are the threshold. Iterations with tactile feedback (left), and without (right).

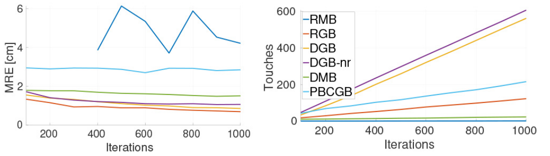


Fig. 12. Right hand reaching for torso—comparison of exploration strategies at 1000 iterations. (Left) MRE. (Right) Number of touches.

by the actual configuration as in the results above) if the difference with the actual configuration is above the 10% threshold for any joint.

To further analyze motor discrepancies, we used histograms of the errors (difference) between target and actual configurations. Each joint was analyzed separately to estimate its range of errors. We also separated the commands that resulted in a tactile feedback, including the discarded ones shown outside the threshold (Fig. 11, left), from the commands that did not provide any feedback (Fig. 11, right).

The error ranges for the commands that did not provide tactile feedback mainly stayed within the 10% threshold (see Fig. 11, right). For the majority of iterations, there is zero, or a very small, error. This was expected as commands that do not produce a tactile feedback are often not in contact at all with the robot. Without collisions, there is little discrepancy between the target and actual commands. For commands that produced tactile feedback, the range of errors is more spread (see Fig. 11, left).

Over all methods and trials, the joint that was more often outside the threshold and participated in the discarding of the target command is the Elbow Roll, followed by the Shoulder Pitch, Shoulder Roll, and the Head Pitch. The Elbow Yaw and Head Yaw had the narrowest ranges of errors, keeping an error close to zero and rarely going over the threshold.

### B. Reaching for the Torso

For each method, we look at the average over all trials of: the MRE and the number of touches through exploration (Fig. 12), and the projection of the skin alongside the distribution of goals generated and the reaching error for each tested taxel (Fig. 13).

Surprisingly, RGB has the lowest MRE after 1000 iterations, close to DGB and DGB-nr. Continuous motion attempt with DGB-nr has very close MRE to DGB with its sporadic motions, contrary to PBCGB that shows an MRE about thrice the value of DGB (Fig. 12, left). DMB displays a low amount of touches, close to RMB, while PBCGB is above RGB but far below both DGB variants which have the highest amount of touches (Fig. 12, right). Looking at the projections (Fig. 13) reinforces the idea that MRE and the number of touches are not good enough metrics by themselves: while somewhat in the middle in MRE and touches, PBCGB shows a high amount of completely unreached taxels, compared to DGB and DGB-nr [Fig. 13(d) and (e)], and surprisingly does not show major improvements even after 15 000 iterations [Fig. 13(f) and (g)]. The difference between DGB and RGB [Fig. 13(d) and (c)] is small despite the gap in the number of touches. Compared to [29], DGB shows a smaller amount of touches and of consistently perfectly reached taxels. DGB-nr has similar or higher reaching errors than DGB, and no taxel is consistently perfectly reached, but has several taxels with mean errors smaller or equal to the distance from the target taxel to its closest neighbor.

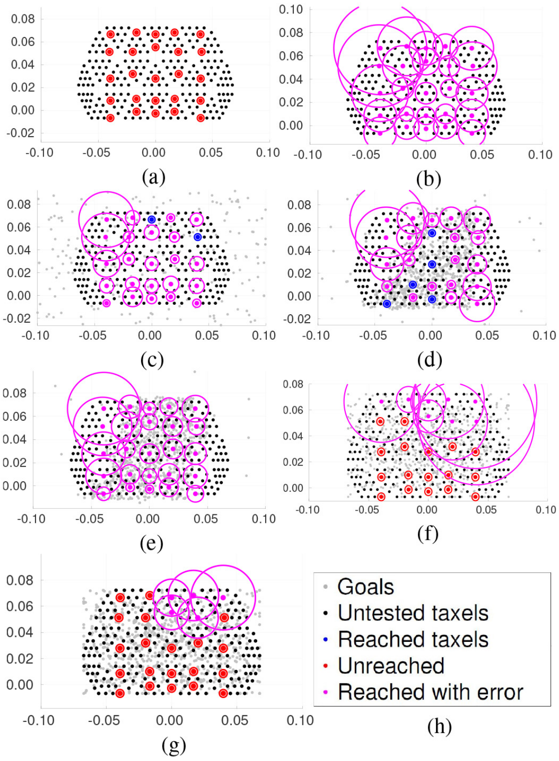
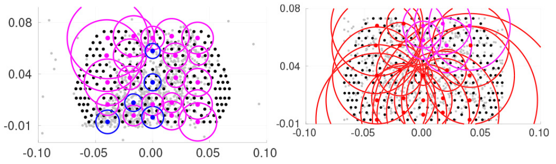


Fig. 13. Right hand reaching for the torso—observation space. See Fig. 9 for details. (a) RMB. (b) DMB. (c) RGB. (d) DGB. (e) DGB—no reset. (f) PBCGB—1000 iterations. (g) PBCGB—15 000 iterations. (h) Legend.

PBCGB takes on average 6 steps to go back to the home posture and 6.5 steps toward an end goal (i.e., there are on average 6.5 goals between two end goals), has 1895 end goals for 15 000 iterations, and it goes back to the home posture 206 times. At 15 000 iterations, it averages 4777±1078 touches.



(a) (b)

Fig. 14. Right hand reaching for the torso—observation space—using 3-D errors. See Figs. 9 and 13 for details. (a) DGB—1000 iterations. (b) PBCGB—15 000 iterations.

TABLE I
MEAN PERCENTAGE OF DISCARDED MOTOR COMMANDS FOR TORSO REACHING, RELATIVE TO 1000 ITERATIONS, EXCEPT PBCGB AT 15 000 ITERATIONS

| RMB | RGB | DGB | DGB-nr | DMB | PBCGB |
|---|---|---|---|---|---|
| 0.3±0.1 | 25.8±9.2 | 18.6±9.5 | 17.7±6.1 | 6.0±5.2 | 2.4±3.9 |

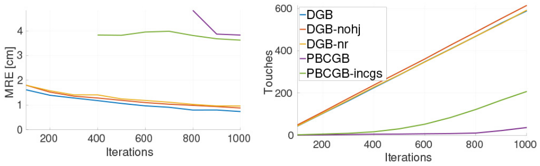


Fig. 15. Right hand reaching for the head—comparison of exploration strategies at 1000 iterations. (Left) MRE. (Right) Number of touches.

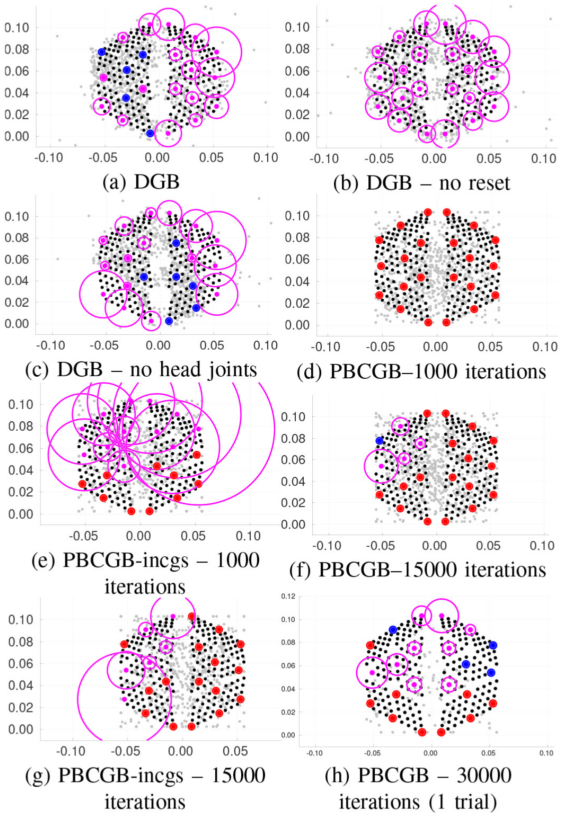
For the torso, we also retrieved the 3-D errors between the end effector and the tested taxels. These errors serve as an external evaluation for unreached taxels during testing (in red in the projections)—they are not used for learning. The 3-D errors are less accurate than the 2-D errors, as shown in Fig. 14 for DGB: reached taxels (in blue) will have an error due to the physical size of the objects even if the finger is in contact with the target taxel, or due to small movements (e.g., sliding) that can happen in the delay between the touch and the recording of coordinates. The 3-D errors in PBCGB are quite high. They seem to intersect around the taxel touched from the home posture, but an analysis of screenshots of the reaching attempts showed that the finger is often away from the skin, rather than around the home posture.

The mean percentage of discarded motor commands is given in Table I.

### *C. Reaching for the Head*

DGB-nohj performs almost as well as DGB with the neck joints enabled with regard to MRE and the number of touches (Fig. 15). The biggest difference is in DGB reaching more accurately the ipsilateral side, while DGB-nohj has higher errors there and instead reaches accurately the opposite side of the face [Fig. 16(a) and (c)]. The goal generation also highlights this shift.

DGB-nr shows a clear increase of the reaching error in the projections [Fig. 16(b)], despite keeping up with the number of touches and an MRE curve that seems close to DGB (Fig. 15). Again, the goal generation is impacted, and more goals are selected on the middle part of the face with no taxels.



(a) DGB (b) DGB – no reset (c) DGB – no head joints (d) PBCGB–1000 iterations (e) PBCGB-incgs – 1000 iterations (f) PBCGB–15000 iterations (g) PBCGB-incgs – 15000 iterations (h) PBCGB – 30000 iterations (1 trial)

Fig. 16. Right hand reaching for the head—observation space. See Figs. 9 and 13 for details. (h) is only one trial that had been run to 30 000 iterations (not a mean of ten trials). (a) DGB. (b) DGB—no reset. (c) DGB—no head joints. (d) PBCGB—1000 iterations. (e) PBCGB-incgs—1000 iterations. (f) PBCGB—15 000 iterations. (g) PBCGB-incgs—15 000 iterations. (h) PBCGB—30 000 iterations (1 trial).

PBCGB only achieves poor performance on the head after 1000 iterations with the default version; no taxel was reached during testing and no error could be measured [Fig. 16(d)]. The version with the increased step length reaches, but with a high error—the intersection of the circles seems to indicate that it mostly reaches for taxels very close to the home posture [Fig. 16(e)]. The number of touches is low for both versions (Fig. 15, right). After 15 000 iterations, the performance improves significantly. There were on average 4778±626 and 5161±402 touches and the MRE dropped to 1 cm. The projections [Fig. 16(f) and (g)] show that reaching for the upper-right part of the head was learned, but the remaining part of the skin is not even reached with error. The goals projected are also mainly present in the center of the middle stripe of the face. This is mainly due to how the algorithm functions, as these include the subgoals, and the middle area is bound to be covered by subgoals when attempting to reach from one side of the face to the other.

We let one of the trials for PBCGB run until 30 000 iterations [Fig. 16(h)] instead of stopping at 15 000. Its projection

TABLE II
MEAN PERCENTAGE OF DISCARDED MOTOR COMMANDS FOR HEAD REACHING. RELATIVE TO 1000 ITERATIONS, EXCEPT PBCGB VARIANTS BASED ON 15 000 ITERATIONS

| DGB | DGB-nr | DGB-nohj | PBCGB | PBCGB-incgs |
|---|---|---|---|---|
| 16.9±6.1 | 17.4±6.7 | 14.4±5.4 | 2.1±3.92 | 0.7±0.9 |

shows that the whole upper part of the head is reached quite accurately, but it is still missing the lower part.

PBCGB takes on average 5.4 steps to go back to the home posture and 6.5 steps toward an end goal; it has 1937 end goals for 15 000 iterations and goes back to the home posture 211 times. The variant with smaller discretization and higher maximum subgoal distance, PBCGB-incgs, takes on average 4.8 steps to go back to the home posture and 2.8 steps toward an end goal; it has 4504 end goals for 15 000 iterations, and goes back to the home posture 452 times.

The mean percentage of discarded motor commands is given in Table II.

## V. Summary and Discussion of Experiments

The results presented in this work are to our knowledge the first attempt to compare the intrinsic motivation framework of the "SAGG-RIAC" type (e.g., [14]) with PBCGB approaches [22]. Furthermore, we study a nonstandard scenario of reaching to own body as opposed to reaching in space. There are a number of complications (see Section III-C and Fig. 5) having mostly to do with the fact that feedback about the learning progress is not available at all times: if no contact on the skin is generated, no error is available and learning as well as "external evaluation" are compromised. For this reason, we complemented the results presentation by reaching error (MRE) with the number of touches achieved and the goal space projection. None of these provide a complete picture, but together they improve our understanding. Additionally, the issue of discrepancy between target and actual joint configurations has to be tackled.

In line with our previous results [29], DGB performed best. Motor space exploration methods—RMB and Motor Babbling with Intrinsic Motivation—performed worse than any Observation space exploration methods—Random or DGB. Beyond Explauto's DGB, the novel PBCGB is intended to learn a model that is both continuous and consistent. While these two key requirements are known to be met for reaching external objects [16], [22], the algorithm only achieves a partial exploration of the touch space. The key limitation of the current setup seems to be the discrete sensor layout with gaps between sensor patches and regions, which makes it hard for the path-based approach to discover the right local movement direction. The difficulty of the requirement for continuous motion is also demonstrated by the degeneration of DGB's performance when generating continuous, "complex" touches in the "no reset" condition.

In summary, this work makes an important contribution in that it puts to test the algorithms developed for reaching in external space on a new problem: learning to reach for the agent's own skin surface. The fact that many reaching trials do not contact the skin and hence provide no training data for learning the inverse model poses a challenge. Additional practical complications come from two facts: 1) self-collisions can occur also between other body parts than the end effector and the target skin region and 2) the skin sensor sheet is not continuous. The algorithms we tested proved to work in this situation but their performance was degraded. Sliding over the skin surface (or "complex touch") provides an opportunity for more efficient learning; however, due to the nature of our setup, it somewhat amplifies the effects of the practical complications mentioned above. A soft skin surface with overlapping tactile receptive fields would be a solution.

## VI. General Discussion and Future Work

First, the issue of motor redundancy should be discussed. The inverse model—from skin space to joint space—was learned directly from the training samples and when using the Explauto methods, it was represented with the nearest neighbor algorithm. While *direct inverse modeling* [52] is prone to the ill-posedness of the general inverse kinematics problem and the averaging over nonconvex solutions sets, our solution circumvents this by performing the exploration in the goal space: alternative solutions exploiting motor redundancy are thus not sought. Additionally, in the nearest neighbor algorithm, no averaging takes place. However, the solution found will in our case be the first solution found; it may thus depend on initialization or chance and may not be the best solution. *Distal learning*, or *learning with a distal teacher* [52], as opposed to direct inverse modeling, is more versatile in that it allows the incorporation of additional constraints to channel the search for the (single) solution. However, while initially, a single solution to a reaching target on the body may suffice, we know that adults are capable of alternative solutions depending on context. Distal learning allows the incorporation of a forward model and inverse model in series. Such a solution is more versatile in that the forward model, which is unique, "disambiguates" between alternatives coming from the, one-to-many, inverse mappings and can check their correctness. Human motor control in the cerebellum may be employing multiple paired forward and inverse models [9] (see the MOSAIC model [53]). Distal learning can thus in principle deal with a redundant system, but the problem is that the motor error is not directly observable [16]. A solution that would allow the agent to find one solution for every reaching target first, but add and keep alternatives later on, remains our future work. The mixed—composite forward-inverse models—can be a solution (see [54] for a survey). In particular, associative memory architectures have been proposed that allow to store several solutions and dynamically switch between them [55]. In this way, it may be possible to construct a *continuous* model of touch that can perform uninterrupted motion throughout complex touches, only flipping to another solution branch if necessary.

Self-touch configurations are also more kinematically constrained than reaching in free space in front of the body and hence the effective motor redundancy is likely lower. This is even more the case for the experiments used here, in which only five DoF of the Nao arm were employed. Other relevant extension points in recent work concern the exploitation of symmetries in the sensorimotor apparatus [56] that can lead to a reduced need for exploration. While previous work developed competence progress-based goal babbling [14] and PBCGB [22] separately, and based on a rather different overall organization of behavior, recent studies have demonstrated their compatibility [24]. The results of the present study regarding accuracy, but also consistency and continuity, suggest that also the (self-)touch domain could benefit from an integration of both approaches that prioritize some paths over others.

Second, the use of the nearest neighbor algorithm for the inverse model representation has to be discussed. It has the following advantages: 1) incremental learning is simple and requires registering pairs from input and goal space only ("lazy learning") and 2) there is no averaging or interpolation of samples (avoiding the problem of nonconvexity of the solution space). The disadvantages are: 1) computational complexity: all experience is stored in memory and upon retrieval—query to the inverse model—time is required to find the nearest neighbor; 2) susceptibility to noise: in our scenario, "phantom" skin activation would be cataloged together with the current joint configuration and contaminate the model; and 3) the mapping will not be smooth: adjoining skin receptors will not necessarily map to nearby joint configurations. Baranes and Oudeyer [14] deemed nonparametric methods (like nearest neighbor) suitable for their problems (including inverse kinematics) and the complexity problem can be mitigated by efficient implementation [57]. Alternative representations of the inverse model could be local regression methods (e.g., Locally Weighted Projection Regression; Sigaud *et al.* [58] for a survey). How such mappings are encoded by the brain is an open question.

Third, the representations of the input and output spaces importantly influence what can be learned and how. In the input or motor space in this work, the actual execution of the movement—initiation, termination, and its dynamics—has not been addressed and such separation of movement preparation and control may not be justified [59]. Mannella *et al.* [27] do consider this aspect and observe, for example, that easy postures are acquired before hard ones. The computational perspective of Cisek could be used as a starting point to add this important dimension [60]. Dynamic Movement Primitives [61] could be employed and possibly allow the incorporation of rhythmic movements; Central Pattern Generators used in [40] are an alternative. Regarding the "skin space," one could come closer to the biological reality by mimicking the nonuniform density of receptors (as done in [25] and [27]). On the representation level, self-organizing maps seem like a natural candidate [27], [62]. The distance metric required for the exploration will then be distorted as is typical for homuncular representations and present an additional challenge. Furthermore, the spaces of torso and head were treated separately in this work. To make them part of the same goal space, a metric connecting them would have to be introduced. Alternatively, they could be separate goals, but within the same exploration framework. Forestier *et al.* [63] or Santucci *et al.* [64] provided possible solutions. Adding the skin space of the effector would bring additional complexity. Finally, the motor and sensory spaces could be treated in a more integrated manner as proposed by Marcel *et al.* [65] who present a mathematical analysis of building a sensorimotor representation of a naive agent's tactile space.

Fourth, it is also worth considering how the task studied here—reaching to the body—differs from reaching in general. As introduced in Section I-A, while learning to touch the body, feedback is not available for most arm postures as the arms are simply in free space. It is also important to consider the role this feedback plays. When the intrinsic motivation frameworks (e.g., [14] and [16]) are applied to reaching in general, an inverse model is being learned and the reaching error is used to monitor learning progress and channel it in different directions. An alternative formulation is offered by Caligiore *et al.* [66], employing reinforcement learning—the agent is rewarded for contacting an object in front of it. In addition, the reward is shaped such that the agent is motivated to reach the final position with a minimum speed, thus mimicking the minimum variance theory [67]. The third component of this model is the equilibrium point hypothesis (e.g., [68]). Thus, this model produces reaching trajectories, while remaining at the level of reaching kinematics (circumventing forces and torques). Remarkably, a number of kinematic variables characterizing the development of reaching in infants is reproduced. However, it is not clear how the reinforcement learning component could be transferred to our case—if reward was delivered for any contact on the skin, the agent would not be motivated to explore the whole skin surface. Related to this is also the notion of goals in general. While in reinforcement learning frameworks the agent is maximizing a sum of future rewards, active goal exploration approaches like the one used here assume that the agent can "imagine" goals—positions on the skin here—and then attempt to achieve them. It remains an open question whether activation of tactile receptors could be interesting enough goals for the infant. Furthermore, and this applies to infant development in general, there is evidence suggesting that infants younger than two years cannot make certain kinds of goal-directed action—they can produce an action directed at a stimulus they see (using an inverse model), but they cannot imagine a stimulus and practice the action [69], [70]. A detailed discussion along with a computational model is presented in [8]. If this counts as evidence for our problem and given that we know the body exploration is mainly occurring during the first year of life, this would exclude the strategies that performed best—goal babbling with intrinsic motivation. However, this is still an open question and it cannot be excluded that infants' use of internal goals may be task- or context-dependent.

It is our ultimate goal to ground the model in biological data. The work of Schlesinger [71], investigating the looking patterns of infants, is an example of such work. In our scenario, there are two concrete ways how we plan to proceed. First, in our study, the robot is learning an inverse model: which motor commands to use to reach to targets on its body. The performance for different body parts and at different stages of development can be compared with behavioral data from infants reaching for vibrotactile stimuli on the body [44]. For example, we should analyze how infants deal with the redundancy of their motor system in this particular case: during different "stages" in their development, do they use the same or distinct configurations to reach for targets on the body? If the latter were the case, the goal exploration strategies that suppress the redundancy of the motor system may not be appropriate. Also, with different initial postures, do infants tend to go to a canonical posture first? There is evidence suggesting that this may be the case in infants [72] and adults learning a new task [73]. A pilot analysis in this direction—following one infant from 4 to 18 months—has been presented here, suggesting that the reaching configurations are quite stereotypical. More quantitative analysis is needed.

Second, statistics obtained from studies observing spontaneous touches to the body in infants [35], [36]—such as how often infants touch particular body parts, in which sequence, etc.—could be fed into the robot simulator to train the inverse model and the results in terms of reaching performance to targets on the body compared with those obtained from the computational exploration strategies. Alternatively, we could aim to model the exploration process itself and obtain similar self-touch statistics as an emergent property. Discovering signatures of curiosity-driven learning in the brain is an active research area [74], employing fMRI [75] or EEG and body states [76]. Only behavioral data poses a greater challenge. With carefully designed experiments, one may be able to discern which cost function the "learning machine" is using [77]. Discriminating spontaneous versus systematic exploration in naturalistic observations (like [35] and [36]) remains to our knowledge an open question.

## Acknowledgment

The authors thank Clement Moulin-Frier for support regarding the Explauto library and the anonymous reviewers for their profound input.

## References

[1] A. Yamaguchi and C. G. Atkeson, "Recent progress in tactile sensing and sensors for robotic manipulation: Can we turn tactile sensing into vision?" *Adv. Robot.*, vol. 33, no. 14, pp. 661–673, 2019.

[2] C. Bartolozzi, L. Natale, F. Nori, and G. Metta, "Robots with a sense of touch," *Nat. Mater.*, vol. 15, no. 9, pp. 921–925, 2016.

[3] S. Luo, J. Bimbo, R. Dahiya, and H. Liu, "Robotic tactile perception of object properties: A review," *Mechatronics*, vol. 48, pp. 54–67, Dec. 2017.

[4] G. Metta *et al.*, "The iCub humanoid robot: An open-systems platform for research in cognitive development," *Neural Netw.*, vol. 23, nos. 8–9, pp. 1125–1134, 2010.

[5] R. M. Bradley and C. M. Mistretta, "Fetal sensory receptors," *Physiol. Rev.*, vol. 55, no. 3, pp. 352–382, 1975.

[6] A. Bremner and C. Spence, "The development of tactile perception," in *Advances in Child Development and Behavior*, vol. 52. Saint Louis, MO, USA: Elsevier, 2017, pp. 227–268.

[7] M. Hoffmann, "The role of self-touch experience in the formation of the self," in *Proc. Develop. Self Workshop IEEE ICDL-EpiRob*, 2017, pp. 1–4.

[8] L. Jacquey, G. Baldassarre, V. G. Santucci, and J. K. O'Regan, "Sensorimotor contingencies as a key drive of development: From babies to robots," *Front. Neurorobot.*, vol. 13, p. 98, Dec. 2019.

[9] M. Kawato, "Internal models for motor control and trajectory planning," *Current Opinion Neurobiol.*, vol. 9, pp. 718–727, Dec. 1999.

[10] A. N. Meltzoff and M. K. Moore, "Explaining facial imitation: A theoretical model," *Infant Child Develop.*, vol. 6, nos. 3–4, pp. 179–192, 1997.

[11] D. Bullock, S. Grossberg, and F. H. Guenther, "A self-organizing neural model of motor equivalent reaching and tool use by a multijoint arm," *J. Cogn. Neurosci.*, vol. 5, no. 4, pp. 408–435, Oct. 1993.

[12] J. Schmidhuber, "A possibility for implementing curiosity and boredom in model-building neural controllers," in *Proc. Int. Conf. Simulat. Adapt. Behav. Animals Animats*, 1991, pp. 222–227.

[13] P.-Y. Oudeyer and F. Kaplan, "What is intrinsic motivation? A typology of computational approaches," *Front. Neurorobot.*, vol. 1, p. 6, Nov. 2007.

[14] A. Baranes and P.-Y. Oudeyer, "Active learning of inverse models with intrinsically motivated goal exploration in robots," *Robot. Auton. Syst.*, vol. 61, no. 1, pp. 49–73, 2013.

[15] G. Baldassarre and M. Mirolli, *Intrinsically Motivated Learning in Natural and Artificial Systems*. Heidelberg, Germany: Springer, 2013.

[16] M. Rolf, J. J. Steil, and M. Gienger, "Goal babbling permits direct learning of inverse kinematics," *IEEE Trans. Auton. Mental Develop.*, vol. 2, no. 3, pp. 216–229, Sep. 2010.

[17] V. Y. Roschin, A. A. Frolov, Y. Burnod, and M. A. Maier, "A neural network model for the acquisition of a spatial body scheme through sensorimotor interaction," *Neural Comput.*, vol. 23, no. 7, pp. 1821–1834, 2011.

[18] M. Rolf, "Goal babbling for an efficient bootstrapping of inverse models in high dimensions," Ph.D. dissertation, Dept. Res. Inst. Cogn. Robot., Bielefeld Univ., Bielefeld, Germany, 2012.

[19] R. M. Ryan and E. L. Deci, "Intrinsic and extrinsic motivations: Classic definitions and new directions," *Contemp. Educ. Psychol.*, vol. 25, no. 1, pp. 54–67, 2000.

[20] A. Baranes and P.-Y. Oudeyer, "R-IAC: Robust intrinsically motivated exploration and active learning," *IEEE Trans. Auton. Mental Develop.*, vol. 1, no. 3, pp. 155–169, Oct. 2009.

[21] K. Narioka and J. J. Steil, "U-shaped motor development emerges from goal babbling with intrinsic motor noise," in *Proc. Joint IEEE Int. Conf. Develop. Learn. Epigenet. Robot. (ICDL-EpiRob)*, Providence, RI, USA, 2015, pp. 55–62.

[22] M. Rolf and J. J. Steil, "Efficient exploratory learning of inverse kinematics on a bionic elephant trunk," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 6, pp. 1147–1160, Jun. 2014.

[23] C. Moulin-Frier, P. Rouanet, and P.-Y. Oudeyer, "Explauto: An open-source python library to study autonomous exploration in developmental robotics," in *Proc. 4th Int. Conf. Develop. Learn. Epigenet. Robot.*, Genoa, Italy, 2014, pp. 171–172.

[24] R. Rayyes, H. Donat, and J. Steil, "Efficient online interest-driven exploration for developmental robots," *IEEE Trans. Cogn. Develop. Syst.*, early access, Jun. 11, 2020, doi: 10.1109/TCDS.2020.3001633.

[25] H. Mori and Y. Kuniyoshi, "A human fetus development simulation: Self-organization of behaviors through tactile sensation," in *Proc. 9th Int. Conf. Develop. Learn. (ICDL)*, Ann Arbor, MI, USA, 2010, pp. 82–87.

[26] Y. Yamada *et al.*, "An embodied brain model of the human foetus," *Sci. Rep.*, vol. 6, Jun. 2016, Art. no. 27893.

[27] F. Mannella, V. G. Santucci, E. Somogyi, L. Jacquey, K. J. O'Regan, and G. Baldassarre, "Know your body through intrinsic goals," *Front. Neurorobot.*, vol. 12, p. 30, Jul. 2018.

[28] D. Caligiore and G. Baldassarre, "The development of reaching and grasping: Towards an integrated framework based on a critical review of computational and robotic models," in *Reach-to-grasp Behavior: Brain, Behavior, and Modelling Across the Life Span*, D. Corbetta and M. Santello, Eds. New York, NY, USA: Routledge, 2018, pp. 335–364.

[29] F. Gama, M. Shcherban, M. Rolf, and M. Hoffmann, "Active exploration for body model learning through self-touch on a humanoid robot with artificial skin," in *Proc. Joint IEEE 10th Int. Conf. Develop. Learn. Epigenet. Robot. (ICDL-EpiRob)*, 2020, pp. 1–8.

[30] A. Piontelli, *Development of Normal Fetal Movements; The Last 15 Weeks of Gestation*. Milan, Italy: Springer-Verlag, 2015.

[31] M. Myowa-Yamakoshi and H. Takeshita, "Do human fetuses anticipate self-oriented actions? a study by four-dimensional (4D) ultrasonography," *Infancy*, vol. 10, no. 3, pp. 289–301, 2006.

[32] S. Zoia *et al.*, "Evidence of early development of action planning in the human foetus: A kinematic study," *Exp. Brain Res.*, vol. 176, no. 2, pp. 217–226, 2007.

[33] P. Rochat, "Chapter 10 hand–mouth coordination in the newborn: Morphology, determinants, and early development of a basic act," *Adv. Psychol.*, vol. 97, no. 1, pp. 265–288, 1993.

[34] P. Rochat, "Self-perception and action in infancy," *Exp. Brain Res.*, vol. 123, nos. 1–2, pp. 102–109, 1998.

[35] B. L. Thomas, J. M. Karl, and I. Q. Whishaw, "Independent development of the reach and the grasp in spontaneous self-touching by human infants in the first 6 months," *Front. Psychol.*, vol. 5, p. 1526, Jan. 2015.

[36] A. DiMercurio, J. P. Connell, M. Clark, and D. Corbetta, "A naturalistic observation of spontaneous touches to the body and environment in the first 2 months of life," *Front. Psychol.*, vol. 9, p. 2613, Dec. 2018.

[37] G. Z. Tau and B. S. Peterson, "Normal development of brain circuits," *Neuropsychopharmacology*, vol. 35, no. 1, pp. 147–168, 2010.

[38] J. Piaget, *The Origins of Intelligence in Children*. New York, NY, USA: Int. Univ. Press, 1952.

[39] A. L. Van der Meer, F. R. Van der Weel, and D. N. Lee, "The functional significance of arm movements in neonates," *Science*, vol. 267, no. 5198, pp. 693–695, 1995.

[40] V. C. Meola *et al.*, "Interplay of rhythmic and discrete manipulation movements during development: A policy-search reinforcement-learning robot model," *IEEE Trans. Cogn. Develop. Syst.*, vol. 8, no. 3, pp. 152–170, Sep. 2016.

[41] L. K. Chinn, M. Hoffmann, J. E. Leed, and J. J. Lockman, "Reaching with one arm to the other: Coordinating touch, proprioception, and action during infancy," *J. Exp. Child Psychol.*, vol. 183, pp. 19–32, Jul. 2019.

[42] L. K. Chinn, C. F. Noonan, and J. J. Lockman, "The human face becomes mapped as a sensorimotor reaching space during the first year," *Child Develop.*, vol. 92, no. 2, pp. 760–773, 2021.

[43] M. Hoffmann *et al.*, "Development of reaching to the body in early infancy: From experiments to robotic models," in *Proc. Joint IEEE Int. Conf. Develop. Learn. Epigenet. Robot. (ICDL-EpiRob)*, Lisbon, Portugal, 2017, pp. 112–119.

[44] J. E. Leed, L. K. Chinn, and J. J. Lockman, "Reaching to the self: The development of infants' ability to localize targets on the body," *Psychol. Sci.*, vol. 30, no. 7, pp. 1063–1073, 2019.

[45] E. Somogyi *et al.*, "Which limb is it? responses to vibrotactile stimulation in early infancy," *Brit. J. Develop. Psychol.*, vol. 36, no. 3, pp. 384–401, 2018.

[46] J. A. S. Kelso, J. J. Buchanan, and T. Murata, "Multifunctionality and switching in the coordination dynamics of reaching and grasping," *Hum. Movement Sci.*, vol. 13, no. 1, pp. 63–94, 1994.

[47] P. Maiolino, M. Maggiali, G. Cannata, G. Metta, and L. Natale, "A flexible and robust large scale capacitive tactile system for robots," *IEEE Sensors J.*, vol. 13, no. 10, pp. 3910–3917, Oct. 2013.

[48] M. Shcherban. (2020). *nao-gazebo-skin*. [Online]. Available: https://github.com/maxymczech/nao-gazebo-skin

[49] A. Baranes and P.-Y. Oudeyer, "Intrinsically motivated goal exploration for active motor learning in robots: A case study," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Taipei, Taiwan, 2010, pp. 1766–1773.

[50] C. Moulin-Frier and P.-Y. Oudeyer, "Exploration strategies in developmental robotics: A unified probabilistic framework," in *Proc. IEEE 3rd Joint Int. Conf. Develop. Learn. Epigenet. Robot. (ICDL)*, Osaka, Japan, 2013, pp. 1–6.

[51] P. Loviken and N. Hemion, "Online-learning and planning in high dimensions with finite element goal babbling," in *Proc. Joint IEEE Int. Conf. Develop. Learn. Epigenet. Robot. (ICDL-EpiRob)*, Lisbon, Portugal, 2017, pp. 247–254.

[52] M. I. Jordan and D. E. Rumelhart, "Forward models: Supervised learning with a distal teacher," *Cogn. Sci.*, vol. 16, no. 3, pp. 307–354, 1992.

[53] M. Haruno, D. M. Wolpert, and M. Kawato, "Mosaic model for sensorimotor learning and control," *Neural Comput.*, vol. 13, no. 10, pp. 2201–2220, 2001.

[54] D. Nguyen-Tuong and J. Peters, "Model learning for robot control: A survey," *Cogn. Process.*, vol. 12, no. 4, pp. 319–340, 2011.

[55] R. F. Reinhart and M. Rolf, "Learning versatile sensorimotor coordination with goal babbling and neural associative dynamics," in *Proc. IEEE 3rd Joint Int. Conf. Develop. Learn. Epigenet. Robot. (ICDL)*, Osaka, Japan, 2013, pp. 1–7.

[56] R. Rayyes, D. Kubus, and J. Steil, "Learning inverse statics models efficiently with symmetry-based exploration," *Front. Neurorobot.*, vol. 12, p. 68, Oct. 2018.

[57] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," in *Proc. 4th Int. Conf. Comput. Vis. Theory Appl. (VISAPP)*, vol. 1, 2009, pp. 331–340.

[58] O. Sigaud, C. Salaun, and V. Padois, "On-line regression algorithms for learning mechanical models of robots: A survey," *Robot. Auton. Syst.*, vol. 59, no. 12, pp. 1115–1129, 2011.

[59] G. Schöner, J. Tekülve, and S. Zibner, "Reaching for objects: A neural process account in a developmental perspective," in *Reach-to-grasp Behavior: Brain, Behavior, and Modelling Across the Life Span*, D. Corbetta and M. Santello, Eds. New York, NY, USA: Routledge, 2018.

[60] P. E. Cisek, "A computational perspective on proprioception and movement guidance in parietal cortex," in *The Somatosensory System: Deciphering the Brain's Own Body Image*. Boca Raton, FL, USA: CRC Press, 2001, pp. 275–297.

[61] A. J. Ijspeert, J. Nakanishi, and S. Schaal, "Learning attractor landscapes for learning motor primitives," in *Advances in Neural Information Processing Systems*. Cambridge, MA, USA: MIT Press, 2003, pp. 1547–1554.

[62] M. Hoffmann, Z. Straka, I. Farkaš, M. Vavrečka, and G. Metta, "Robotic homunculus: Learning of artificial skin representation in a humanoid robot motivated by primary somatosensory cortex," *IEEE Trans. Cogn. Develop. Syst.*, vol. 10, no. 2, pp. 163–176, Jun. 2018.

[63] S. Forestier, R. Portelas, Y. Mollard, and P.-Y. Oudeyer, "Intrinsically motivated goal exploration processes with automatic curriculum learning," 2017. [Online]. Available: arXiv:1708.02190.

[64] V. G. Santucci, G. Baldassarre, and M. Mirolli, "GRAIL: A goal-discovering robotic architecture for intrinsically-motivated learning," *IEEE Trans. Cogn. Develop. Syst.*, vol. 8, no. 3, pp. 214–231, Sep. 2016.

[65] V. Marcel, S. Argentieri, and B. Gas, "Building a sensorimotor representation of a naive agent's tactile space," *IEEE Trans. Cogn. Develop. Syst.*, vol. 9, no. 2, pp. 141–152, Jun. 2017.

[66] D. Caligiore, D. Parisi, and G. Baldassarre, "Integrating reinforcement learning, equilibrium points, and minimum variance to understand the development of reaching: A computational model," *Psychol. Rev.*, vol. 121, no. 3, pp. 389–421, 2014.

[67] C. M. Harris and D. M. Wolpert, "Signal-dependent noise determines motor planning," *Nature*, vol. 394, no. 6695, pp. 780–784, 1998.

[68] A. G. Feldman, "Once more on the equilibrium-point hypothesis (λ model) for motor control," *J. Motor Behav.*, vol. 18, no. 1, pp. 17–54, 1986.

[69] U. M. H. Klossek, J. Russell, and A. Dickinson, "The control of instrumental action following outcome devaluation in young children aged between 1 and 4 years," *J. Exp. Psychol. Gen.*, vol. 137, no. 1, pp. 39–51, 2008.

[70] B. Kenward, S. Folke, J. Holmberg, A. Johansson, and G. Gredebäck, "Goal directedness and decision making in infants," *Develop. Psychol.*, vol. 45, no. 3, pp. 809–819, 2009.

[71] M. Schlesinger, "Investigating the origins of intrinsic motivation in human infants," in *Intrinsically Motivated Learning in Natural and Artificial Systems*. Heidelberg, Germany: Springer, 2013, pp. 367–392.

[72] N. E. Berthier, R. K. Clifton, D. D. McCall, and D. J. Robin, "Proximodistal structure of early reaching in human infants," *Exp. Brain Res.*, vol. 127, no. 3, pp. 259–269, 1999.

[73] M. Rohde, K. Narioka, J. J. Steil, L. K. Klein, and M. O. Ernst, "Goal-related feedback guides motor exploration and redundancy resolution in human motor skill acquisition," *PLoS Comput. Biol.*, vol. 15, no. 3, 2019, Art. no. e1006676.

[74] J. Gottlieb and P.-Y. Oudeyer, "Towards a neuroscience of active sampling and curiosity," *Nat. Rev. Neurosci.*, vol. 19, no. 12, pp. 758–770, 2018.

[75] M. J. Gruber, B. D. Gelman, and C. Ranganath, "States of curiosity modulate hippocampus-dependent learning via the dopaminergic circuit," *Neuron*, vol. 84, no. 2, pp. 486–496, 2014.

[76] A. Appriou, J. Ceha, E. Law, P.-Y. Oudeyer, and F. Lotte, "Towards measuring states of curiosity through electroencephalography and body sensors responses," in *Proc. Journées CORTICO*, 2019, p. 1.

[77] J. G. Cashaback, H. R. McGregor, A. Mohatarem, and P. L. Gribble, "Dissociating error-based and reinforcement-based loss functions during sensorimotor learning," *PLoS Comput. Biol.*, vol. 13, no. 7, 2017, Art. no. e1005623.

# Appendix F

# Robotic homunculus: Learning of artificial skin representation in a humanoid robot motivated by primary somatosensory cortex

Authors' version of:

Hoffmann, M., Straka, Z., Farkas, I., Vavrecka, M., and Metta, G. (2018). Robotic homunculus: Learning of artificial skin representation in a humanoid robot motivated by primary somatosensory cortex. *IEEE Transactions on Cognitive and Developmental Systems* 10 (2): 163-176.

DOI: `https://doi.org/10.1109/TCDS.2017.2649225`
Youtube video: `https://youtu.be/8aiSHzJrJLE`

Author contributions: The contribution of M. Hoffmann was 32%.

# Robotic homunculus: Learning of artificial skin representation in a humanoid robot motivated by primary somatosensory cortex

Matej Hoffmann, *Member, IEEE,* Zdeněk Straka, Igor Farkaš, Michal Vavrečka,
and Giorgio Metta, *Senior Member, IEEE*

***Abstract*—Using the iCub humanoid robot with an artificial pressure-sensitive skin, we investigate how representations of the whole skin surface resembling those found in primate primary somatosensory cortex can be formed from local tactile stimulations traversing the body of the physical robot. We employ the well-known self-organizing map (SOM) algorithm and introduce its modification that makes it possible to restrict the maximum receptive field (MRF) size of neuron groups at the output layer. This is motivated by findings from biology where basic somatotopy of the cortical sheet seems to be prescribed genetically and connections are localized to particular regions. We explore different settings of the MRF and the effect of activity-independent (input-output connections constraints implemented by MRF) and activity-dependent (learning from skin stimulations) mechanisms on the formation of the tactile map. The framework conveniently allows one to specify prior knowledge regarding the skin topology and thus to effectively seed a particular representation that training shapes further. Furthermore, we show that the MRF modification facilitates learning in situations when concurrent stimulation at non-adjacent places occurs (“multi-touch”). The procedure was sufficiently robust and not intensive on the data collection and can be applied to any robots where representation of their “skin” is desirable.**

***Index Terms*—artificial skin, self-organizing maps, somatosensory cortex, tactile sensor, humanoid robot.**

## I. Introduction

The somatotopic representations discovered in the primary motor and somatosensory cortices of primates [1], [2] have attracted extensive attention because of their unquestionable importance in “interfacing” the brain with the body. Somatotopy of these brain areas is often visualized in form of “homunculi” (“little men”) that facilitate presentation to a wider audience and stimulate researchers to investigate the origin of the correspondence of the cortical representations with the motor and somatosensory systems. The pioneering work of Leyton, Sherrington, Penfield and others was later refined using more accurate techniques; the single “somatosensory homunculus” of [3], Fig. 1 (A), for example, was replaced by four individual full homunculi in the areas 3a, 3b, 1, and 2 of the anterior parietal cortex. The two areas fed primarily by tactile (rather than proprioceptive) inputs are 3b and 1, with area 3b being the most ”primary”. Detailed somatotopic organization in area 3b of the macaque based on the results of [4] is shown in Fig. 1 (C).

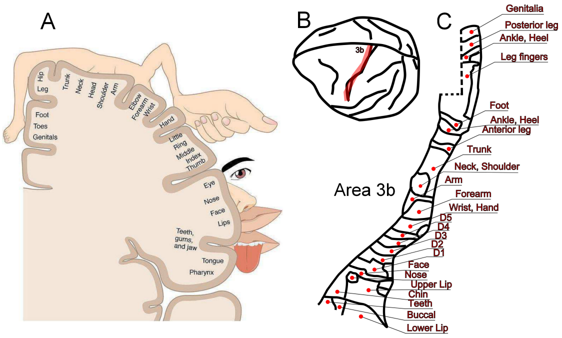


Fig. 1. **Somatosensory homunculus.** (A) Famous somatosensory homunculus of man after Penfield and Rasmusen [3]. Reprinted from [5] under a CC BY license, with permission from OpenStax College, original copyright 2013. Download for free at http://cnx.org/contents/29cade27-ba23-4f4a-8cbd-128e72420f31@5. (B) Dorso-lateral view of the brain showing the location of area 3b. (C) Organization of the representations of body surface in area 3b of the cynomolgus macaque. Area 3b is shown “unfolded” from the central sulcus and medial wall of the hemisphere. Cortical areas activated by designated body surfaces are outlined. Representations of individual digits of the hand are outlined and numbered ($D_1$ corresponding to thumb, $D_5$ to little finger); the dashed line indicates the region along the medial wall where portions of the representation are contained in the cortex on the medial wall of the hemisphere. Redrawn and simplified after Fig.1, [4].

The formation of these representations has become an important topic in the “nature vs. nurture” debate. Two extreme positions are constituted by the *activity-independent* view, which claims that establishment of topographic maps is a result of patterning intrinsic to the nervous system and does not require specific neural activity, and the *activity-dependent* or self-organization view, which attributes a key role to the patterns of neural activity in the process of somatosensory neural circuits development. This idea was elaborated by Crair [6] who concludes: “Where the development of a particular neural circuit lies in this continuum probably depends on a number of factors, including the presence of neural activity in

The contributions of M. Hoffmann and Z. Straka to this work were equal.

M. Hoffmann and G. Metta are with iCub Facility, Istituto Italiano di Tecnologia, Genova, Italy; e-mail: {name.surname}@iit.it

M. Hoffmann and Z. Straka are with Dept. Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic; e-mail:{name.surname}@fel.cvut.cz

I. Farkaš is with Centre for Cognitive Science, Department of Applied Informatics, Faculty of Mathematics, Physics and Informatics, Comenius University in Bratislava, Slovak Republic.

M. Vavrečka is with Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic.

the developing neurons, the particular stage of development involved, and whether there is competition between different pools of neurons for postsynaptic target territory." While this statement applies to central nervous system development in general, we will focus on the somatosensory cortex, in particular on the representation of cutaneous inputs (i.e. inputs originating from the skin; in the remainder of this article, we will use "tactile" to refer to these inputs, because this term is more compatible with the terminology in robotics). The interplay of genetically determined and activity-dependent factors encompasses the whole ascending pathway—specifically the posterior column–medial lemniscal pathway that carries "fine touch". Somatotopy is present in the ascending fibers and at all "relay stations": the dorsal root ganglion, the medulla, the VPL nucleus of the thalamus, and finally in the neocortex (with area 3b considered the most "primary"). The activity-independent topographical arrangements may come from molecular gradients between specific areas. Vanderhaeghen et al. [7], for example, provide evidence that certain proteins act as within-area thalamocortical mapping labels in rat's S1 and affect topography as well as the relative size of individual areas. Conversely, others have amassed evidence for the activity-dependent factors in map formation (e.g., [8], [9]). The interplay of these two factors will be central to our experiments on a humanoid robot with artificial skin.

There have been different models of topographic map formation proposed. Some of them contain considerable neurobiological detail: von der Malsburg and Willshaw [10] modeled the axon growing mechanism between two neural sheets. Pearson et al. [11] studied breaking up of their "model cortex" into clusters, applying the neuronal group selection theory. Models that choose a higher abstraction level include the dynamic field theory [12] and self-organizing maps (SOMs) [13]–[15]. These computational models were restricted to small simulated "skin patches" and controlled stimulation. Some researchers moved beyond bottom-up single modality processing models to multisensory settings (Pitti et al. [16] studied visuo-somatosensory alignment in the superior colliculus) or fully embodied sensory-motor settings: Kuniyoshi and colleagues (e.g., [17]) developed a fetal simulator with the aim to investigate the effect of its embodied interaction in the uterine environment on early neural development. Some of these works specifically addressed somatosensory cortex development (e.g., [18], [19] using Hebbian learning and denoising autoencoder, respectively).

With the advent of robotic tactile sensing technologies [20]–[24], learning the skin representationy gains practical importance: robots are in need of such representations of their skin surface that can be used in control (e.g. in collision isolation and reaction) or in tactile human-robot interaction (see [25] for a survey; [26] for a recent implementation on the iCub humanoid robot). Denei et al. [27] provide an overview and present a method of obtaining a 2D tactile map, which can be advantageous for control purposes, from a previously obtained 3D skin mesh (using [28] or [29], for instance). McGregor et al. [30] developed a method based on information distance (ANISOMAP) that is able to reconstruct 3D tactile surface (in a topological, not metric, sense) from uninterpreted tactile data. In these approaches, every tactile sensor is typically represented—without compression of the input space. The SOM algorithm, on the other hand, possesses the vector quantization property in that it allocates a smaller number of output representatives ("neurons") in an optimal fashion with respect to the density of input vectors (resembling the cortical representations that reflect the innervation density of different skin parts as well as the stimulation frequency). Pugach et al. [31] used the SOM to learn a representation of the surface of a conductive material that did not have any discrete tactile sensors; instead, the stimulus location and pressure on a continuous sensor surface were reconstructed using electrical impedance tomography and the voltage matrices thus obtained were fed as inputs to the SOM.

Our overarching research approach is the so-called synthetic methodology [32]: understanding natural phenomena by realizing them in artificial systems and, at the same time, seeking how to turn the artifacts into applications. First, the biologically motivated line of this work consists in using a baby humanoid robot with tactile arrays covering most body parts to investigate the possibility of somatotopic map formation from physical stimulation of the skin. With the map from the primate somatosensory cortex as an approximate target, we explore the effect of parameters of the SOM algorithm, initial conditions, constraints, and input data properties on the output map. To this end, we introduce a SOM modification that makes it possible to restrict the maximum receptive field (MRF) size of neuron groups at the output layer—mimicking the activity-independent "patterning" of the cortex. At the same time, mirroring the organization of primary somatosensory cortex is only one possible target. The embodiment of the humanoid robot is obviously not identical with primates—in particular, the characteristics and placement of tactile sensors and the "neural system" and its constraints are different—, therefore, we also study the behavior of the algorithm in less constrained settings and analyze representations that emerge from the contingencies intrinsic to the robot. To the best of our knowledge, this is the first investigation in this scale and in a real robot. The output of this work, the different "robotic tactile homunculi", will be used in subsequent research on the iCub that targets the development of multimodal body representations (see [33] for a survey of the notion of body schema in robotics, [34] for an account of the iCub learning a peripersonal space representation using the artificial skin, and [35] for learning a proprioceptive representation).

Second, pursuing the "useful artifacts/algorithms" line, the modified SOM algorithm proposed is surely applicable more generally in engineering settings. The presented procedure was found sufficiently robust and not very intensive on the data collection and can thus be applied to any robots where representation of their "skin" is desirable. The fact that the desired map organization can be easily specified is particularly convenient, as it allows to seed the representation exploiting prior knowledge about the skin spatial arrangement (which is often available) and/or consider other criteria on the properties of the output map that may be dictated by how this representation will be used in subsequent processing. Furthermore, we show how the MRF modification improves SOM learning in case of "multi-touch".

This article is structured as follows. The Materials and Methods with detailed descriptions of the setup and the algorithms used comes immediately after the Introduction, followed by Results, and finally Conclusion, Discussion, and Future Work.

## II. Materials and Methods

### A. iCub Robot and Artificial Skin

The iCub is an open-source platform for research in cognitive robotics [36]. Its mechanical design is detailed in [37]. The iCub was recently equipped with an artificial pressure-sensitive skin covering most body parts [38]. There are skin patches on the torso (440 taxels), arms (380 taxels on each upper arm), forearms (230 taxels each), palms (44 each), and fingertips (12 taxels per fingertip). In total, these comprise 1928 tactile elements. In this work, we use the skin on the right half of the upper body—a schematic and photo of the skin layout on the trunk and one arm is depicted in Fig. 2. With the exception of palms and fingertips, the skin is composed of triangular modules, each of them hosting 10 taxels. The taxels respond proportionally to the pressure applied to them. However, in this work, we restricted ourselves to binary values (0 $\sim$ inactive, 1 $\sim$ active) only. The data were sampled at 50 Hz.

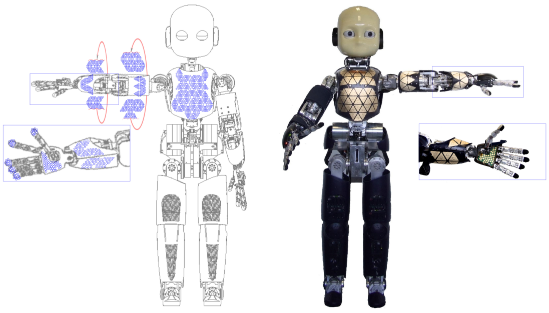


Fig. 2. **Artificial skin on the iCub.** (left) Schematic illustration of the layout of skin patches on one half of the upper body. The patches covering arm and forearm that would not be visible in this view have been unfolded. (right) Photograph of the real robot in analogous posture and exposed skin on corresponding body parts.

### B. Training Data

*1) Synthetic Training Data:* In order to analyze the properties of the algorithm under controlled conditions, synthetic data sets were used in the first step. The training data were generated on a simulated skin with a rectangular grid—nodes of the grid representing taxels. Skin activations were simply represented by a matrix $S$.

$$S(i,j) = \begin{cases} 1 & \text{for a stimulated taxel} \\ 0 & \text{for a non-stimulated taxel} \end{cases} \tag{1}$$

The simplest method of generating a single touch, with $m = numRows$ and $n = numCols$, would be:

1) Randomly choose a taxel $t_1$ (with a position $(i,j)$, $i \in \{1,..,m\}$, $j \in \{1,..,n\}$).
2) Find all adjacent taxels $\{t_2, t_3, ..., t_o\}$ to the taxel $t_1$ chosen in the previous step. If the taxel $t_1$ is not on the edge of the skin, the number of adjacent taxels is eight, otherwise the number of adjacent taxels is lower.
3) For all $(k,l) \in \{1,..,m\} \times \{1,..,n\}$ set
$$S(k,l) = \begin{cases} 1 & \text{if pos.}(k,l) \text{ matches one of } \{t_1,..,t_o\} \\ 0 & \text{otherwise} \end{cases}$$

However, application of this algorithm would lead to a nonuniform distribution of taxel activations, with those at the edges less frequently activated (there is a smaller number of adjacent taxels that could serve as the locus of simulated touch). In order to guarantee a uniform distribution of taxel activations, the grid was circumscribed by a row/column of "virtual taxels" along all edges. Each of the "virtual taxels" could be chosen as the central taxel $t_1$ of a touch. The actual activations calculated according to the pseudocode above were confined to the original dimensions of matrix $S$ though. The code implementing this is available under S1_Code (function `createTouches2`) in [39]. Multi-touches were generated by independently iterating the algorithm above, giving rise to an activation matrix for each touch. These were then summed and finally a ceiling function was applied to each element to ensure it is bounded by 1.

*2) Tactile Stimulations in Real Robot:* Whenever individual skin parts were stimulated, the experimenter was sliding with the tip of a single finger, mostly the thumb, along the skin surface, stimulating on average between 6 and 12 taxels at a time (for the fingertips, only 3). In some regions, such as on the "edges" of the arm (see Fig. 2), there are small gaps between individual skin patches. In one place, the fabric covering the skin has also a stitch on the surface. In these locations, two fingertips were sometimes used to ensure co-activation of the regions along the boundaries. The stimulation sequence was random—to the extent that this could be ensured by a human experimenter.

To study multi-touch on the robot, the torso was used and two experimenters were sliding along the torso with one thumb each, giving rise to the double-touch data set that will be used in the second part of Section III-A. The experimenters were trying to move independently and to spend roughly equal times at different locations. The total stimulation time was around 9 minutes, giving rise to 28000 data points—see $VideoMultitouch.mp4$ at [39] for an illustration.

Finally, to generate the training data for the complete "tactile homunculus", stimulations from the whole skin surface were necessary. Individual skin parts were stimulated as described above. However, in addition, the data had to contain co-activations of abutting skin parts in order to provide input material to the SOM algorithm to extract the topological relationships. Compared to humans, the skin parts in the robot are less continuous—joints, for example, are lacking skin coverage. To mitigate this effect, special stimulations that generated activations along the borders of neighboring skin parts (such as adjacent fingertips, fingertips and palm, palm and forearm etc.) were necessary. The robot was put into configurations where the skin parts in question were not too far apart. Even so, the gaps did not allow for the co-stimulations to be generated using a single object and two hands had to be used instead. An illustration of how this was done is provided in $VideoStimulationIllustration.mp4$ and $VideoStimulationIllustrationDesktopRecording.mp4$ at [39]. The number of data points per skin part that formed the training set is detailed in Table I. Less than half an hour of stimulation time was necessary for a complete half of the robot's body. The logic behind the particular choice of ratios will be explained under III-B.

TABLE I
STIMULATION FREQUENCY OF INDIVIDUAL SKIN PARTS FOR HALF OF THE ROBOT'S BODY

| | Nr. taxels | Nr. data points | stimulation time [s] |
|---|---|---|---|
| individual digits | 5×12 | 9000 | 180 |
| palm | 44 | 6300 | 126 |
| forearm | 230 | 15700 | 314 |
| upper arm | 380 | 15700 | 314 |
| torso | 440 | 22000 | 440 |
| | 1154 | 68700 | 1374 |
| adjacent digits | | 6700 | 134 |
| palm+digits | | 1000 | 20 |
| palm+forearm | | 1000 | 20 |
| forearm+upper arm | | 2000 | 40 |
| upper arm+torso | | 3000 | 60 |
| | | 13700 | 274 |

### C. Self-organizing Map with Maximum Receptive Field Size Setting (MRF-SOM)

The classical version of the self-organizing feature (Kohonen) map was described in [40], [41]. We use the variant with the dot product application to determine the best matching unit (winner) for a given input: DP-SOM (rather than the variant with Euclidean distance—the motivation for this choice will be explained below). We follow the formalization of [41], in which both variants are presented. The same formula—dot product—is used to determine the activation of output neurons after learning.

The classical SOM, as its name suggests, relies purely on self-organization and learns from the inputs in an unsupervised way. While this may be ideal in many situations, in some other cases, there may exist prior knowledge or constraints that should be applied to steer the adaptation in specific directions. In our case, which deals with the problem of mapping the whole-body skin surface to a 2-dimensional output sheet, there is no perfect solution. Evolution of primate nervous systems has led to one particular solution to the problem that reappears, with variations, in different species. With this coarse topology as our target, we were seeking a modification of the SOM algorithm that allows to impose some constraints on the output layer topology. Our proposed solution is loosely inspired by the synaptic connections in the ascending somatosensory pathway, which are not all-to-all, but confined to specific regions, with overlaps to neighboring regions (see e.g., [42]). In a similar way, we have developed a solution to impose "masks" on weight vectors between the input and output layers, allowing to nullify certain connections. Conversely, each output neuron has a mask of 1's to certain regions of the input space, thus defining a maximum possible extent of its receptive field—hence the name MRF-SOM, SOM with Maximum Receptive Field size setting. After learning, each neuron will specialize on a specific part of the input space, which will necessarily lie within the MRF constraint.

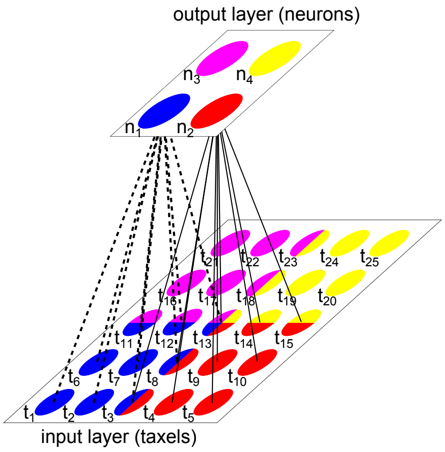


Fig. 3. **Illustration of MRF-SOM.** Four output neurons, $n_1, n_2, n_3, n_4$, are shown at the top. At the bottom layer, there are 25 inputs, taxels $t_1, ..., t_{25}$. The color code and the weight vectors (weights shown only for $n1, n2$) mark the maximum receptive field size setting of the output neurons. See text for details.

We will illustrate how this is implemented with the help of Fig. 3. There are four output neurons: $n_1, n_2, n_3, n_4$. At the input layer, there are 25 taxels arranged on a square grid. Let us further assume that we want to define a 3×3 maximum receptive field (MRF) for each output neuron, pointing to respective corners of the input grid. This is schematically illustrated with color codes of the neurons and their respective RFs on the input grid; taxels with multiple colors indicate overlapping MRFs of the neurons. The way this is implemented is through the weight vectors: for each output neuron, a mask is applied to the elements of its weight vector. So, taking neuron $n_1$, for example, the mask applied to its weight vector is specified in Eq. 2:

$$\begin{aligned}\mathbf{mask_1} = [&1, 1, 1, 0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 0, 0,\\ &0, 0, 0, 0, 0, 0, 0, 0, 0, 0]\end{aligned} \qquad (2)$$

This is also schematically depicted in Fig. 3: the dashed lines correspond to weight vector components of $n_1$ that have a "1" component in the mask. The missing lines correspond to the nullified connections. Analogous pattern is shown for $n_2$. The mask will be reapplied to the weight vector at each iteration of the algorithm—elements outside the MRF may have been subject to adaptation and hence become non-zero until the mask is applied.

Now we will show how we modified the learning of the DP-SOM. Let's have a DP-SOM with $k$ neurons. Each neuron has its own weight vector. We will denote by $\mathbf{m}_i$ the weight vector of the neuron $i$. The winner neuron (indexed by $c$) is determined by using the dot product as $c = \arg\max_i\{\mathbf{m}_i \cdot \mathbf{x}(t)\}$, where $\mathbf{x}(t) \in \{0,1\}^l$ is an input vector whose dimension in our case equals the number of taxels. Adaptation of the weight vectors of the DP-SOM during learning is then realized by the rule in Eq. 3 below (the "bell curve" neighborhood as per II.B and the dot product formulation from Section II.F in [41]):

$$\mathbf{m}_i(t+1) = \frac{\mathbf{m}_i(t) + h_{ci}(t)\mathbf{x}(t)}{\|\mathbf{m}_i(t) + h_{ci}(t)\mathbf{x}(t)\|} \tag{3}$$

where $\|.\|$ denotes the Euclidean norm, $h_{ci}(t) = \alpha(t)\exp(-\|\mathbf{r}_i - \mathbf{r}_c\|^2/(2\sigma^2(t)))$ is the Gaussian neighborhood function, where the learning rate $0 < \alpha(t) < 1$ decreases monotonically with time, $\mathbf{r}_i, \mathbf{r}_c \in \mathcal{R}^2$ are the vectorial locations on the output grid, and $\sigma(t)$ corresponds to the width of the neighborhood function, which also decreases monotonically with time. Adaptation in MRF-SOM is realized by these steps:

i.

$$\mathbf{m}_i^{'} = \mathbf{m}_i(t) + h_{ci}(t)\mathbf{x}(t) \tag{4}$$

ii.

$$\mathbf{m}_i^{'} = \mathbf{m}_i^{'}.*\ \mathbf{mask}_i \tag{5}$$

iii.

$$\mathbf{m}_i(t+1) = \frac{\mathbf{m}_i^{'}}{\|\mathbf{m}_i^{'}\|} \tag{6}$$

where the vector $\mathbf{mask}_i \in \{0,1\}^l$ is the mask of the neuron $i$. Application of Eq. 5, using component-wise multiplication of two vectors, sets the elements of the weight vector $\mathbf{m}_i^{'}$ corresponding to taxels that are not connected with neuron $i$ to zeros. Everything else in the MRF-SOM algorithm is the same as in the DP-SOM algorithm.

The choice of the DP-SOM as opposed to the Euclidean distance version was primarily motivated by the winner selection step. For every neuron in the MRF-SOM, the weights outside its MRF are nullified as per Eq 5. Thus, the input vector components outside a neuron's RF do not affect the winner neuron determination. However, this would not be the case in the Euclidean distance version. Furthermore, it is a characteristic feature of our data set that the majority of input vector components are 0; the dot product computation in this case is faster.

*1) Implementation and Parameters of MRF-SOM Training:* A freely available SOM toolbox [43] was used. Training is implemented in the `som_seqtrain` function. However, this is the Euclidean distance variant of the algorithm. Therefore, we performed necessary modifications for the dot product version as well as added the maximum receptive field size setting as specified above (MRF-SOM).

The following input parameters were used: a hexagonal lattice in the shape of a sheet, a Gaussian neighborhood function with initial radius of 5 and final equal to 1, and the learning rate decayed from the initial value 0.5 according to $\alpha(t) = a/(t+b)$ with suitably chosen parameters $a$ and $b$. The remainder of the parameters followed default settings; for more information use the online documentation of the SOM toolbox [44]. In addition, the `MRF` input parameter was added. Rows of the parameter `MRF` express the maximal possible ranges of RFs of the neurons. The code used is available under S2_Code in [39].

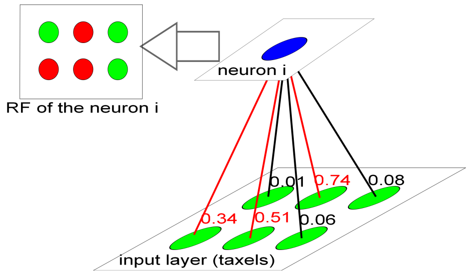


Fig. 4. **Receptive field determination from weight vectors.** For a given threshold, here 0.3, all taxels connected with a neuron with a weight exceeding the chosen threshold are marked as belonging to the RF of the neuron.

### *D. Receptive Fields and Visualization of Learned Maps*

Given the relation of our study with somatotopic maps from biology, it is the receptive fields of neurons in the learned maps that are crucial. That is, for each neuron of the output map, we need to know the region of skin (the set of taxels) whose stimulation evokes that neuron's response. Two different techniques were employed in this work.

*1) Weight Vector Components Exceeding the Threshold:* The first method of receptive field determination is straightforward: for each neuron, its weight vector is inspected and all the taxels that are connected with the neuron with a weight exceeding a certain threshold are marked as belonging to its RF. This is illustrated schematically in Fig. 4. However, this method is rather a top-down shortcut that gives only a quick overview. Furthermore, the threshold needs to be set empirically and depends on the weight vector size.

*2) Biomimetic RFs Determination:* The second method we used was inspired from biology and the way RFs are determined using microelectrode recordings in electrophysiology, where localized tactile stimulations are applied and neuronal responses recorded. In a similar vein, we emulated this procedure by replaying a testing set that consisted of stimulations (single localized stimulations, not multi-touch) similar to the ones used for training and recorded the winner neurons. A pseudocode of this "bottom-up" algorithm is given below. Basically, every neuron has its RF ($rf$), which is initially an empty set. As the algorithm iterates through the stimulations, winning neurons enlarge their RFs by including the taxels stimulated at a given time. An example of a map visualized using this method is Fig. 11.

Pseudo-code of the "biomimetic RF determination" algorithm:

*Input*: $Mtest$ (test set with touch stimulations), threshold $K$

```
1) Init:
    $\mathbf{M}_i = [0,0,..,0]$ for all $i \in \{1,2,..,N\}$, where the length of all $\mathbf{M}_i$ is equal to the number of taxels and $N$ is number of neurons.
    $rf_i = \emptyset$ for all $i \in \{1,2,..,N\}$
2) For each touch $\mathbf{tch}$ from $Mtest$
    determine the winner $w$ for touch $\mathbf{tch}$
    $\mathbf{M}_w = \mathbf{M}_w + \mathbf{tch}$   i.e. increment number of taxels stimulations from $\mathbf{tch}$ for winner
3) For each neuron $i$ in the grid
    add taxels from $\mathbf{M}_i$ that exceed threshold $K$ to $rf_i$
    plot taxels from $rf_i$ with red color, others with green color
```

*3) Heuristic Visualization of Learned Maps:* To obtain a visualization of the learned maps, we preferred the "biomimetic" method. However, sometimes, there was a fair amount of neurons that ended up with empty RFs after application of this method—they never won after any stimulation from the testing set. Yet, these neurons did learn to represent some parts of the skin, as revealed by analysis of their weight vectors. In this case, the first method—looking at weight vector components exceeding the threshold—was applied in a second step, allowing to assign RFs to the remaining neurons. A heuristic threshold was applied. In this way, we could generate visualizations where each neuron can be colored according to the body part(s) it represents, as will be shown in Figs. 14, 16, 17.

### *E. Topology Preservation Measure with External Distance Metric (TPMEDM)*

To complement visual inspection of the learned maps and to allow for quantitative comparison of different settings of the algorithm, numerical measures assessing the quality of learned maps are desirable. Various measures have been proposed to numerically assess the organization of the trained SOMs (for an overview, see [45] and references therein). For instance, the topographic product [46] considers only the codebook vectors (weight vectors) after learning and measures the distances of $k$ nearest neighbors of each neuron in the output space as well distances between the prototypes in the input space, eventually combining them into a single number that summarizes the quality of the topology preservation. However, the input space we are dealing with here renders this method inappropriate due to the particular nature of distances in the input space. Although a skin patch is a 2D surface (embedded in a 3D space), our input space is very different: it has as many dimensions as there are taxels and every dimension can take only discrete values $\{0,1\}$. Imagine a 3×3 skin patch with taxels $t_1, ..., t_9$ shown in Fig. 5.

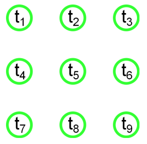


Fig. 5. Schematic illustration of a 3×3 skin patch with 9 taxels.

The input will simply be a 9-dimensional vector of activations $\mathbf{A}$, like in Eq. 7.

$$\mathbf{A} = (A_{t_1}, A_{t_2}, A_{t_3}, A_{t_4}, A_{t_5}, A_{t_6}, A_{t_7}, A_{t_8}, A_{t_9}) \quad (7)$$

It is apparent that using the Euclidean distance formula, different "atomic" touches (activation of only one taxel) will have identical distances from each other—no matter where they lie on the skin. For example, stimulation of taxel 1 ($\mathbf{A}^1$, Eq. 8) will have the same distance from the neighboring taxel 2 (Eq. 9) and from a "far away" taxel 9 (Eq. 10)—all distances being equal to $\sqrt{2}$.

$$\mathbf{A}^1 = (1,0,0,0,0,0,0,0,0) \quad (8)$$

$$\mathbf{A}^2 = (0,1,0,0,0,0,0,0,0) \quad (9)$$

$$\mathbf{A}^9 = (0,0,0,0,0,0,0,0,1) \quad (10)$$

In case of multiple concurrent taxel stimulations, the distance will get smaller if the stimulations overlap on some taxels. However, the set of Euclidean distances computed for the given input data will be very discrete ("step-like"), rather than continuous, so it cannot give satisfactory results.

Another commonly applied measure, topographic error, does not rely on any distance measurements. For every input data point, it determines the first and second best-matching units and checks whether these are adjacent on the output map lattice. This information is then aggregated and normalized. This measure is more suitable in our situation and we experimented with it.

However, we finally decided to employ a quality measure that directly measures the main objective of the representation: how the actual skin surface topology is preserved in the "cortical sheet"—the output lattice. That is, we decided to utilize information that is external to the algorithm itself, namely the actual distances between the taxels on the skin. This information is not available to the SOM algorithm—only indirectly through the co-stimulations of adjacent taxels present in the input data. However, it is available to us (at least for the simulated skin and for individual skin parts, like the torso) and we will thus directly use it to assess the quality of learned maps. Our measure also uses the RF concept, as defined in II-D above.

The measure we are proposing, Topology Preservation Measure with External Distance Metric (TPMEDM), basically evaluates whether the taxels composing RFs of adjacent neurons on the cortical sheet are also close to each other on the skin surface. This is schematically illustrated in Fig. 6.

The TPMEDM measure is evaluated as follows:

```
1) Determine RFs $rf_i$ for all neurons $i = 1,2,..,N$, using the biomimetic method specified in II-D.
2) For all adjacent neurons $i, j$ on the output map lattice, make a union of their RFs $rf_{i,j} = rf_i \cup rf_j$
        For all pairs of taxels $k, l \in rf_{i,j}$, compute the taxel distance using the external distance function.
3) Return the mean taxel pair distance.
```

Experimentation with this measure on our data proved that it is superior to the topographic product and topographic

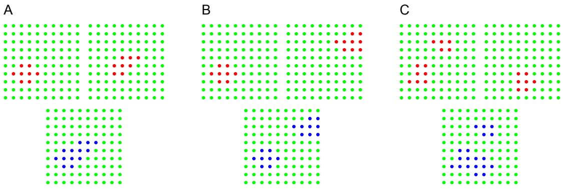


Fig. 6. **Union of two receptive fields.** Top: In each case (A, B, C), the red taxels in the two panels represent hypothetical RFs of two adjacent neurons. Bottom: Blue taxels represent the union of the RFs from the top row. The distances between every taxel pair in this union forms the basis of the TPMEDM. (A) RFs of adjacent neurons are close to each other and partly overlap. (B) RFs are distant. (C) One of the RFs is not compact. Since RFs of adjacent neurons are closest in the case A, the mean of distances of all pairs of blue taxels in union of both RFs is smaller than in the cases B and C.

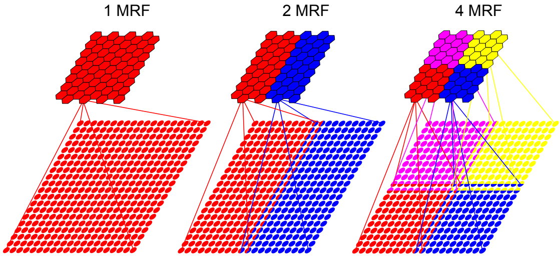


Fig. 7. **Illustration of three variants of MRF setting for simulated skin experiments.** From left to right: 1 MRF, 2 MRF, 4 MRF. There are 8×8 output neurons shown at the top and 20×20 inputs (simulated taxels) at the bottom. The color code and the span of weight vectors mark the maximum receptive field size of every output neuron area. Taxels with multiple colors mark the overlap of maximum receptive fields.

error measures (but keeping in mind that it utilizes external distance information) and in most cases it matches well with the visual assessment of learned maps. The code implementing this measure can be found under S1_Code in [39].

## III. Results

The Results section is split into two parts. The first part is devoted to learning correct topology of a skin surface using the SOM from input data that contain multiple concurrent stimulations. A modification of the SOM algorithm that mitigates the problems resulting from "multi-touch" will be presented and tested in a series of experiments on a simulated skin surface and later on real data coming from concurrent stimulation of the robot torso. The second part presents a series of experiments where the SOM algorithm together with the proposed modification is used to learn a single representation of the skin surface of one half of the robot's body—giving rise to the "robotic homunculus" analogous to the lateralized representations in primate somatosensory cortex.

This section will feature both figures with actual results (such as learned maps) and schematics showing the algorithm settings, for example. For better orientation of the reader, all "Results figures" captions will be preceded with "Results – ". All Tables report results.

### A. Toward More Realistic Stimulation – Learning From Multi-touch

*1) Multi-touches on Simulated Skin:* In these experiments, we simulated different numbers of concurrent stimulations on a skin model and investigated their effect on map formation. Multi-touch in general degrades the quality of learned maps, because the standard SOM algorithm is not able to naturally cope with multiple concurrent stimulations: it treats them as a single point in the input space, resulting in learning (weight vector adaptation) in undesired directions. However, the SOM modification presented here, MRF-SOM (see the corresponding Section II-C under Methods for details), can be employed to mitigate this effect. For each neuron at the output, a maximum possible extent of its receptive field (RF) is prespecified; subsequently, each neuron will learn to be sensitive to a subset of this maximum region of input space. The maximum receptive field (MRF) regions with only a partial overlap will then ensure that activations will remain grossly localized and hence interference between far away input space regions will be reduced.

The skin model had a size of 20×20 taxels (tactile elements, modeling individual pressure sensors in the robot). Training data consisted of 100 000 $k$-touches, with $k \in \{1, 2, 4, 6, 8\}$ fixed for each training set. Stimulations of taxels followed a uniform distribution; for details of the generation see Section "Synthetic training data" under II. The MRF-SOM had a size of 8×8 neurons and was trained for 24 epochs. Additional parameters and details of the implementation can be found in Section II-C. Three variants of the MRF setting were studied. In the first case, each neuron's MRF contained all taxels (1 MRF), which is equivalent to unmodified SOM (the MRF setting having no effect). This is illustrated schematically in Fig. 7, left panel. In the second case (2 MRF), if neuron $i$ is on the left half of the map, then its MRF contains only taxels from the left part of the skin. Two rows of taxels in the center of the skin are shared by neurons from left and right halves of the map. The third case (4 MRF) is similar to the second but the neurons and their MRFs are divided in four partially overlapping squares. The overlap is necessary in order to smoothly connect the representations at the boundaries.

An illustration of the results is depicted in Fig. 8 (right) for the most challenging input type: 8 concurrent stimulations. For visualization, we used a biologically inspired method of determining the RFs of individual neurons – please see Section "Biomimetic RFs from simulated skin stimulation" under II-D. The left panel (1 MRF, i.e. standard SOM without MRF) shows the difficulty the standard SOM is facing with this input: practically all neurons learn discontinuous RFs (red areas in every subplot). Conversely, the problem is significantly mitigated in the case of four MRFs – see the right panel, where the majority of neurons have a single continuous RF in the input space.

Space limitations will not permit us to graphically demonstrate all the combinations of input types (number of concurrent touches) and algorithm settings (number of MRF). Therefore, we developed a custom measure of the quality of learned maps: TPMEDM (see Section II-E for details), which

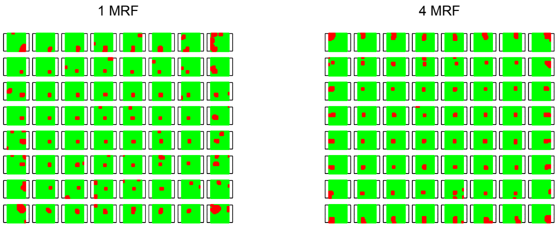


Fig. 8. **Results – Learning from 8-touches in simulated skin. 1 MRF (left) vs. 4 MRF (right) settings.** The 8×8 matrix is the lattice of the output neurons. Every element (subplot) then depicts a miniature version of the simulated skin, in which the set of red taxels represents the receptive field of the corresponding neuron. If there is more than one red area per subplot, it means the neuron's RF is not continuous.

TABLE II

MULTI-TOUCH ON SIMULATED SKIN. QUALITY OF LEARNED MAPS IN TERMS OF TPMEDM FOR DIFFERENT COMBINATIONS OF INPUT (1-TOUCH TO 8-TOUCH) AND MRF SETTING, USING THE $mean \pm std$ NOTATION TO SUMMARIZE 10 RUNS OF THE ALGORITHM. LOWER VALUES CORRESPOND TO BETTER MAPS.

| | 1 MRF | 2 MRF | 4 MRF |
|---|---|---|---|
| 1-touch | 2.97 ± 0.01 | 3.01 ± 0.00 | 3.03 ± 0.01 |
| 2-touch | 3.13 ± 0.15 | 2.99 ± 0.04 | 3.03 ± 0.16 |
| 4-touch | 5.86 ± 0.99 | 4.17 ± 0.41 | 3.43 ± 0.10 |
| 6-touch | 6.80 ± 0.80 | 4.23 ± 0.26 | 3.61 ± 0.13 |
| 8-touch | 6.48 ± 1.77 | 4.39 ± 0.28 | 3.57 ± 0.14 |

correlates with the visual intuition regarding the topology preservation. For every combination of stimulation type and MRF setting, 10 repetitions of the learning algorithm were run, using a different training set and initial weight settings. Aggregate results in terms of TPMEDM between the runs are shown in Table II (including the standard deviation) and Fig. 9: the lower the TPMEDM value, the better the quality of the map. It is evident that the topology preservation capability of standard SOM (1 MRF) degrades rapidly in the case of 4 and more concurrent touches. This is significantly improved already if two MRFs are used; 4 MRFs make the degradation in performance even for 8-touch very small. The apparent non-monotonicity in some of the values along the $k$-touch axis lies within the standard deviation intervals.

The data and code related to this section are available under S1_Data_and_Code in [39].

*2) Multi-touch on the iCub Torso:* In this section we verify our findings from the simulated skin on the real robot. The largest single skin surface, the torso with 440 taxels (see Section II-A) was chosen and stimulated by either one experimenter (1-touch or single touch) or two experimenters (2-touch or double touch). Please recall that single touch stands for a single stimulated area of a couple of adjacent taxels (around 12 on average in this case) at a time; double touch corresponds to two such independent, disjoint areas. The procedure gave rise to 28000 samples and is described in more detail in Section II-B, "Tactile stimulations in real robot", with a link to a video.

Similarly to the previous section, the output layer of the map had 8×8 neurons and the map was trained for 25 epochs, with the same parameter settings. Four MRF settings were tested: 1 (i.e. standard SOM), 2, 4, and 8. This is illustrated in Fig. 10. The MRF regions were overlapping at their boundaries.

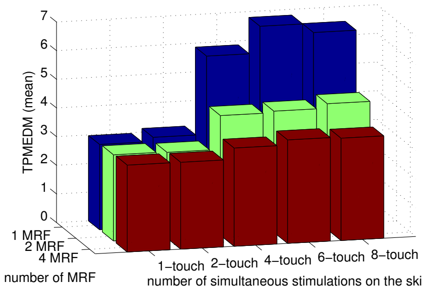


Fig. 9. **Results – Multi-touch on simulated skin – graphical representation of the means from Table II** Lower values correspond to better maps in terms of TPMEDM. The plot reveals the drop in performance of standard SOM (∼ 1 MRF) when faced multi-touch and how this effect is counterbalanced by the use of MRF.

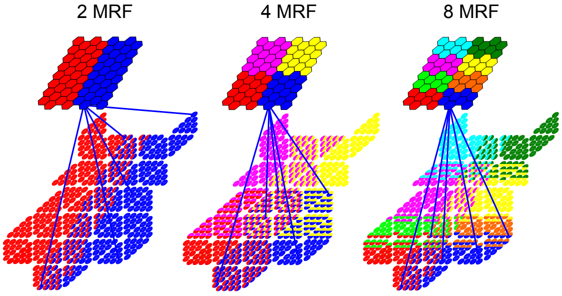


Fig. 10. **Illustration of three variants of the MRF setting for multi-touch on iCub torso.** From left to right: 2 MRF, 4 MRF, 8 MRF (1 MRF not shown). There are 8×8 output neurons at the top; 440 taxels of torso skin in a 2D arrangement (before its attachment on the 3D robot torso) are at the bottom. Color code and weight vector span mark the MRF setting of output neurons; taxels with multiple colors signify MRF overlap.

Analogously to our findings on the simulated skin array, multi-touch makes it more difficult for a SOM to capture the input space topology also for real data sets. This is illustrated in the left panel of Fig. 11. Some neurons have learned discontinuous RFs; furthermore, the overall topology of the torso skin is not well represented in the map. Conversely, "pre-parcellation" of the space into coarse, partially overlapping regions using the MRF setting significantly improves the situation. The case of 8 MRF is shown in the right panel: the RF sizes are comparable to the 1 MRF case, but the topology preservation is clearly superior.

Aggregate results for all combinations of stimulation type and MRF settings using TPMEDM (see II-E) are depicted in Table III and Fig. 12. The quality of learned maps clearly degrades when the training set contains double touches. The MRF setting successfully mitigates this effect and performance correlates with the number of MRFs used.

Compared to the results from the simulated skin, shown

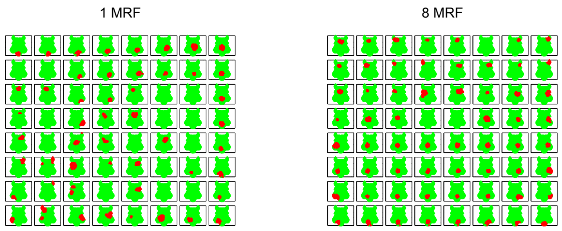


Fig. 11. **Results – Learning from 2-touches in iCub torso skin. 1 MRF (left) vs. 8 MRF (right) settings.** Every element of the 8×8 matrix depicts a miniature version of the torso skin array, in which the set of red taxels represents RF of the corresponding neuron—according to the position on the lattice.

TABLE III
**SINGLE TOUCH AND 2-TOUCH ON ICUB TORSO SKIN.** QUALITY OF LEARNED MAPS IN TERMS OF TPMEDM FOR DIFFERENT COMBINATIONS OF INPUT (1-TOUCH OR 2-TOUCH) AND MRF SETTINGS, USING THE $mean \pm std$ NOTATION TO SUMMARIZE 10 RUNS OF THE ALGORITHM. LOWER VALUES CORRESPOND TO BETTER MAPS.

| | 1 MRF | 2 MRF | 4 MRF | 8 MRF |
|---|---|---|---|---|
| 1-touch | 28.99 ± 0.51 | 28.65 ± 0.18 | 28.63 ± 0.12 | 28.08 ± 0.08 |
| 2-touch | 40.40 ± 1.17 | 37.06 ± 1.22 | 33.99 ± 0.60 | 30.65 ± 0.69 |

in Table II and Fig. 9, double touch appears to present significantly higher difficulties in the case of real data. (Note that the comparison can take into account only the differences within a data set; the absolute values of TPMEDM cannot be compared between data sets, because the measure utilizes the actual distance between taxels, but the scale of the two skin arrays is different.) We attribute this to the overall less favorable statistical properties of the real data sets, mainly due to the data collection procedure. Despite every effort of the experimenters to stimulate all taxels uniformly, a plot of the distribution of taxel activations within a data set reveals that this was not the case, with number of stimulations per taxel ranging from around 400 to around 2500 stimulations and the portions of skin at the borders being significantly less stimulated (for the case of double touch see S1_Fig.eps at [39]). This nonuniformity will naturally be reflected in the learned map. Furthermore, there is a difference between the synthetic and the real training set: the "real" touches, unlike synthetic ones, are not completely independent (even if each of them is made by a different experimenter). These problems are in a sense inherent to data sets collected by humans in this way. However, our results show how the problem can be largely alleviated using the proposed MRF-SOM—if approximate topology of the surface to be mapped is known beforehand.

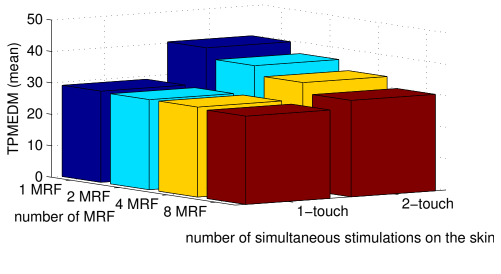


Fig. 12. **Results – Single touch and 2-touch on iCub torso skin – graphical representation of the means from Table III**. Lower values correspond to better maps in terms of TPMEDM. The plot reveals the drop in performance on 2-touch and how this effect is counterbalanced by the use of MRF.

The data and code related to this section are available in S2_Data_and_Code in [39].

### *B. Robotic Tactile Homunculus*

In previous sections, we studied the effects of different stimulation and algorithm parameters on a problem where all inputs were located on an essentially 2D input space (the torso of the real robot is not exactly planar, but can be approximated as such) and then represented by a SOM with 2D topology on the output layer. There was thus a relatively clear optimum, which the algorithm with its properties (optimal representation of input space, topology preservation) could come close to. In this section, the goal is to represent tactile sensors of the "whole", or significant parts of, the body surface in the same output sheet with 2D topology. Some skin parts are locally planar, but already relatively simple parts, such as an upper arm, present a problem to the standard rectangular lattice, due to the neighborhood relation on opposite sides of the sheet (there is no beginning and end of the skin around the arm). This could be mitigated by using a toroidal lattice, but for the case of the whole skin surface, all body parts cannot be possibly arranged on a 2D sheet preserving all neighborhood relations. Thus, some discontinuities are inevitable.

To test our algorithm, we have targeted one particular type of solutions to this problem, namely the one resembling those present in the primary tactile cortex of primates – see Fig. 1. Primary representations in the brain are always lateralized; therefore we focus on building representations of the right half of the body only (including the trunk, which is present in both halves). Another striking factor of cortical representations is the magnification of certain body parts, which is primarily attributed to different degrees of skin innervation. Our target roughly corresponds to the part of area 3b from the trunk to the digits (fingers). This region is highlighted in Fig. 13 (A), along with the correspondences on the macaque monkey body (B).

Of course, the robot and its artificial skin differs from the monkey in numerous aspects. First, our version of the robot does not have capacitive skin on the face or the legs. Second, the skin is composed of identical modules, which corresponds to constant innervation density (with the exception of the palm and fingertips that use different technology, but still with a similar density). Moreover, there is a much larger absolute number of taxels on larger body parts: 440 on torso, 380 on upper arm, compared to mere 104 on palm and fingers

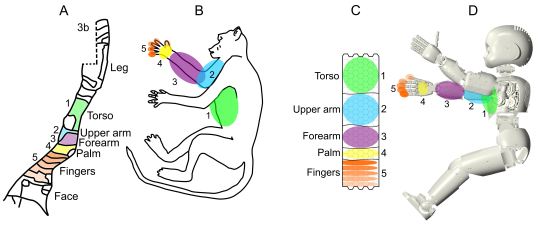


Fig. 13. **Representation of tactile body surface in monkey and robot.** (A-B) Simplified representation of selected body parts in area 3b of macaque monkey. Numbers and color code mark the correspondences between the cortical areas and skin surface on the body parts that will be modeled using the iCub robot. Redrawn and adapted after [4]. (C-D) Schematics of analogous situation in the robot – approximate target for the SOM algorithm.

(see II-A for details). A "uniform" stimulation of the robot's surface would thus give rise to very different proportions of the homunculus. Therefore, to be able to influence the number of neurons devoted to different body parts at the output layer, we will manipulate the stimulation frequency of individual skin parts. Roughly inspired by these proportions, but taking the actual taxel counts per body part in the robot also into account, we chose corresponding ratios of stimulation time, and hence also the number of training data points, per skin part. The details of the stimulation procedure (i.e., touching the robot's skin), including a video and a table showing the exact numbers, are in Section II-B. In all experiments in this section, the output map lattice was 24×7 (to mimic somewhat the elongated shape of the tactile homunculus in the cortex) and the map was trained for 25 epochs; all remaining parameters are specified in Section II-C.

*1) Homunculus Learning without MRF Setting:* In the first step, we have applied the standard SOM algorithm (dot product version, DP-SOM) without additional constraints (no MRF setting) using the training set as described above. Five complete independent runs of the algorithm were executed; the results of three of them are depicted in Fig. 14. We want to make the following points regarding the distribution of RFs on these maps: First, there is high variability in the outcome of different runs of the algorithm resulting in very different topology of the learned map. Sometimes, some skin parts' representations fill a compact "strip" across the whole longer dimension of the map; sometimes, they extend along this longer dimension. Torso, palm, and fingers' portions of the map remain always compact (in the right-most map, palm and fingers not neighboring though), whereas the forearm and upper arm representations are often separated into multiple disjoint areas. This could be attributed to the fact that they are composed of multiple skin patches wrapped around a toroidal or smooth cuboidal shape, which is far from planar, and perhaps also the fact that they are centrally located in the chain and thus may be pulled by their neighbors to different directions. Second, the size occupied by different body parts in the learned map also varies: for example, from 64 to 80 neurons devoted to the torso or from 26 to 38 for the forearm. Third, as anticipated, the outcome departs considerably from the arrangement present in the biological maps (area 3b – cf. Figs. 1 and 13). The results confirm the intuition that the problem of fitting the whole skin surface onto a 2D sheet is under-constrained and there is no perfect solution. It seems that there are multiple local extremes that the algorithm may converge to. The convergence properties could improve if significantly larger training set was available and slower learning rate was applied. However, it seems impossible that self-organization alone would bring about the same representations of palm and finger regions in the map as it is in the somatosensory homunculus, for example.

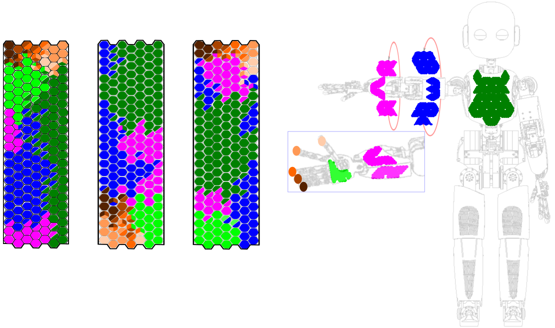


Fig. 14. **Results – Learning from tactile stimulation on right side of robot body with standard DP-SOM.** The three panels on the left depict the maps (24×7 neurons) after learning as a result of three runs of the algorithm on the same training set. The visualization, which colors the maps according to the RFs of individual neurons, is the result of the "Heuristic visualization of learned maps" procedure described in Section II-D. Neurons with multiple colors signify that the taxels belonging to their RF are part of more than one skin part. The right panel shows body parts that correspond to the colors in the maps. Supporting material illustrating how the visualization was arrived at for the map in the middle and on the right is shown in S2_Fig.svg and S3_Fig.svg respectively at [39].

*2) Homunculus Learning with MRF Setting:* In order to address the shortcomings of the maps learned in the previous section, here we employ the MRF setting (see II-C) to steer the self-organizing process in desired directions. That is, unlike Section III-A, where we showed how MRF improved SOM adaptation when the training data contained multiple disjoint stimulations, here only single stimulations were used, but MRF-SOM is exploited in order to ensure coarse topology of the representation as well as approximate proportions of areas devoted to individual skin parts.

The overall layout is depicted in Fig. 13, (C-D), illustrating the desired sequence of areas and their rough proportions. This gross layout is then translated into specific MRF settings: one variant is shown in Fig. 15. The MRF region of the output map dedicated to a specific skin part spans that skin part and an adjacent region of the neighboring skin part.

An example of a learned map with these settings is in Fig. 16. The left panel shows the RFs of the upper-most 49 neurons of the lattice—the region devoted to the torso—demonstrating reasonable coverage of the area as well as appropriate topology preservation. The whole map—in the middle panel—testifies good preserva-

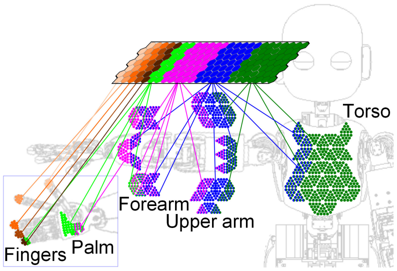


Fig. 15. **Detailed MRF setting for learning tactile homunculus.** The colors and lines ascending from the individual skin parts to the neuronal sheet schematically illustrate the MRF settings. Every skin part has its "dedicated" area in the neuronal sheet—dark green for torso, blue for upper arm, pink for forearm, light green for palm, and tones of orange for fingertips. In addition, the skin areas bordering with another skin part belong to the MRF of the adjacent area of neurons as well—as illustrated by the color code. The palm and finger areas are an exception to this rule: tighter MRF settings were used here to warrant that the learned map will have topology analogous to area 3b. The particular order of digits, with little finger adjacent to the palm representation, would otherwise not emerge from running the algorithm on the training set.

tion of the "desired layout" (MRF setting) and the actual learned topology. An illustration of the activations in the learned map during tactile stimulation is provided in $VideoStimulationsAndMapActivations.mp4$ at [39]. This map meets the criteria of obtaining a representation that is—on a certain level of abstraction—faithful to the biological blueprint, but adapted to the robot, and will be used in further work where a biologically motivated representation of the robot's tactile inputs is necessary.

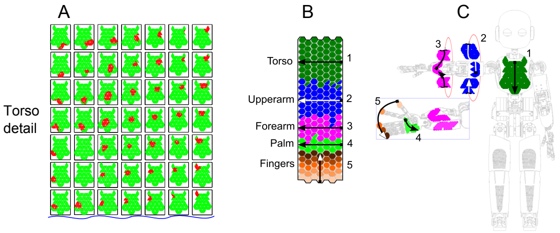


Fig. 16. **Results – Learning from tactile stimulation with MRF-SOM.** (A) Top section of the output map – 7×7 neurons with miniature depictions of the torso skin; red taxels mark RFs of the corresponding neuron (visualization using the "biomimetic RF determination method"; see Section II-D). (B) Visualization of the whole map using two-stage "heuristic visualization". Neurons with multiple colors indicate that the taxels composing their RF belong to more than one skin part. (C) Body parts with color code corresponding to the map. Inspired by the visualization in Fig. 13, the arrows illustrate how the coarse orientation of individual skin parts is represented in the map. For example, the top-to-bottom direction of the torso skin was roughly translated into right-to-left in the map. Supporting material illustrating how the visualization was arrived at is shown in S4_Fig.svg at [39].

*3) Simulating Lesion of One Body Part:* In light of the apparent stringency of the underlying MRF constraints outlined in Fig. 15, the result presented in Fig. 16 may appear to be somewhat unsurprising. We have decided to explicitly test the degree of plasticity that is still present in the network with detailed constraints. To this end, we have simulated a lesion of the upper arm skin by pruning three quarters of the corresponding training set segments where this part was stimulated. The learned map in Fig. 17 demonstrates that despite the stringent MRF constraints, the neighboring skin parts (torso and forearm) significantly expanded their representations at the expense of the upper arm region. Furthermore, even the palm representation could take advantage of the situation and seize new territory. The data and code related to this section are available in S3_Data_and_Code at [39].

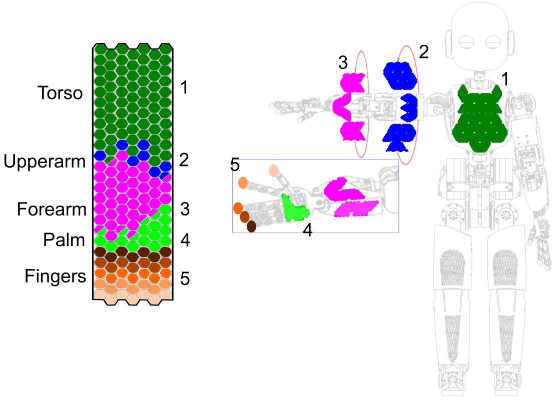


Fig. 17. **Results – Learning from tactile stimulation with MRF-SOM and simulated lesion of upper arm.** The same settings and visualization as in Fig. 16 were used, but 3/4 of upper arm stimulations were pruned in the training set.

## IV. Conclusion, Discussion, Future Work

In this article, we presented work studying how a humanoid robot with sensitive skin could learn a topographic representation of its body surface from experience—by receiving tactile stimulations all over its artificial skin. Having stimulated the robot's skin on the upper body for about half an hour in total, we studied the settings of the well-known self-organizing feature map (SOM, or Kohonen map) algorithm that are required to channel the learning into a target representation resembling the one known from the primate cortex. To this end, we proposed a modification of the standard SOM algorithm (MRF-SOM) that allows to prespecify certain, partially overlapping, receptive fields of the output layer neurons. This guarantees that certain proportions as well as the sequence of the represented areas can be specified *a priori*. This may, on one hand and at a high level of abstraction, mimic the known connectivity from the ascending somatosensory pathway with divergent connections (e.g., [42]), but it mainly constitutes a simple but practical tool to guide SOM learning in desired directions. We also show that even if relatively specific "seeding" of the map is applied, the network does retain sufficient plasticity to suppress representation of a lesioned region of the input space. Furthermore, the standard SOM algorithm is not able to cope with multiple concurrent stimulations (such as simultaneous touch on different body parts): it treats them as a single point in the input space, resulting in weight adaptation in undesired directions. The proposed modification significantly increases the performance in this case. At the same time, the proposed MRF-SOM algorithm is easily portable to other robots that feature some form of artificial skin array (e.g., [47], [48]; [20], [21] for reviews) and can be deployed to tailor the map learning process to any criteria specified by the user (such as the availability of prior knowledge of skin arrangement or the desired properties of the output layer). Finally, the new TPMEDM measure quantifying the quality of the observed maps, which relates the distance of adjacent taxels peripherally to their separation on the generated map (see Section II-E for details), is another contribution of this work.

The goal was not to obtain a mathematically optimal representation (which would inevitably have to be 3-dimensional, e.g., [49]), but rather one motivated by the primary representations of tactile (cutaneous) receptors in primate brains. This is one of the well-known "somatosensory homunculi", concretely the one of Brodmann area 3b. If the cortical sheet is unfolded, one can imagine a 2-dimensional grid of neurons with a somatotopic arrangement of receptive fields, mimicking the spatial arrangement of the cutaneous receptors on the body surface, but with inevitable discontinuities resulting from the dimensionality reduction (the skin forms a continuous structure in three dimensions). There is thus no perfect solution to this problem in terms of topological or topographical criteria and the one adopted by biological systems is a result of various historical, evolutionary, anatomical (nerves from different body parts reach spinal or later thalamic nuclei at different locations) and other constraints.

As already discussed, the level of chosen abstraction regarding the putative biological processes in operation was very high. Some of the decisions as to the model parameters were dictated by the platform we used. For example, the artificial skin of the iCub responds to sustained pressure only, which may be said to grossly emulate the response of Merkel disk receptors (slowly adapting mechanoreceptors present in human skin). In terms of receptive field size, the situation may be somewhat comparable: (i) Although individual Merkel disk receptors are much smaller than the taxels in the robot, the dorsal root ganglion cells innervating superficial skin layers receive input from 10–25 Merkel disk receptors, giving rise to a receptive field spanning a circular area with a diameter of 2–10 mm ( [50], p. 435), which is comparable to the taxel diameter of 4 mm in the robot; (ii) Cortical neurons have larger receptive fields than sensory afferents, spanning for example half a fingertip or areas of several centimeters in diameter on less densely innervated body parts (see [4], for example). This is again roughly comparable to the situation in the robot after learning, where RF sizes also range from parts of a fingertip to fractions of the palm surface (roughly 1–2.5 cm in diameter) to several triangular skin modules on other body parts (1.5–4 cm in diameter, for example). However, there is a number of important differences that limit the biological plausibility of our setup. First, the skin mechanics and the receptor embedding in the robot and in biology is most probably completely different (see [51] for a 3D finite element model of the finger distal phalange and [52] for a review of prosthetic electronic skin.) Second, with mere 1154 receptors on the half of the robot upper body and only 24×7 neurons on the output layer, the numbers are significantly smaller than in the biological realm. Third, the overlap and redundancy of the representation are largely limited, compared to what is expected from the biological counterpart. Fourth, we have only emulated one receptor type (Merkel disk, isotropic response only in our model), while it has been hypothesized recently that "touch is a team effort": the submodalities of touch (slowly and rapidly adapting mechanoreceptors, Pacinian afferents) interact. Thus the traditional perspective relying on submodality segregation and receptive field mapping using artificial, submodality specific stimuli is limited—the alternative being natural, multimodal, stimuli and analysis of neuron firing based on their function [53]. Fifth, any attentional mechanisms were out of our scope—but see [54], for example. Finally, regarding the artificial neural circuitry employed, it has to be stated that the "relay stations" of the ascending pathway with additional functionality like inhibitory surround were ignored and a direct mapping from the "receptors" to the "cortex" was learned instead (similarly to [10]–[12]; [55] used a 3-layer network).

The SOM algorithm itself has been shown to give rise to receptive field structures that resemble those of real neurons (e.g., [56]). One decision on our part has been that we have worked with binary inputs only. However, we have conducted an empirical comparison with continuous data (both simulated and real from the iCub torso), both variants leading to very similar maps after training. A report summarizing our results $BinaryVsContinuousStimuli.pdf$ is available at [39]. Another feature that is probably at odds with putative neuronal mechanisms is the global supervisory mechanism in SOM that determines the winning neuron during learning. It could be replaced by recurrent interactions between neurons though, which was already present in von der Malsburg's model [10] and later in the LISSOM model (Laterally Interconnected Synergetically Self-Organizing Map; [57]) or the recent GCAL variant (Gain Control, Adaptation, Laterally connected; [58]). It is possible that these algorithms may perform better when faced with multi-touch stimulations—this needs to be tested in the future. Another variant of the algorithm that is relevant in this situation is the DSOM (Dynamic SOM; [59]), in which the time-dependent learning function (learning rate and neighborhood radius decreasing over time) was replaced by a time-invariant function, triggering learning as soon as inputs that lack a close representative are encountered. This would be a way of achieving life-long learning in the robot and could be one of the possible implementations leading to the well-known plasticity (reorganization capability) of the cortical maps (see e.g., [12], [13], [55] for models dealing with somatosensory cortex). This constitutes another direction of future work.

In summary, as a model of somatosensory (tactile, more precisely) cortex development, the work presented operates at a high level of abstraction and has admittedly important limitations. However, its contribution to the neurosciences and cognitive sciences should be best viewed as a building block, part of a larger project that aims at embodied modeling of primate body and peripersonal space representations. Our effort parallels that of Kuniyoshi et al. dealing with foetal development (e.g., [17]), but focuses on early postnatal development and uses a real robot as opposed to simulation. The maps representing the robot's skin that originated in this work will be used in ongoing work that studies the development and operation of multimodal (tactile, proprioceptive, visual) body representations. The development of proprioceptive representations is studied in parallel [35] as well as learning from visuo-proprioceptive-tactile associations about peripersonal space [34]. At the same time, these developments may, first, set the ground for future refinement of the work presented here. In particular, self-touch (as developed for the iCub in [60]) holds great promise as an autonomous multimodal body schema learning tool. Second, with several modules in place, the possibilities for behavioral testing of the learned representations—accuracy of gazing at or removal of vibrating stimuli, for example—will be open. At the same time, this work is relevant for robotics, in particular for physical human-robot interaction: robots with artificial skin and representations thereof are more aware of the full occupancy of their bodies, leading to safer interaction with their surroundings. Finally, all the data and code used in this work are available at [39] and we would be happy to assist other researchers in using it.

## Acknowledgment

MH was supported by the Swiss National Science Foundation (www.snf.ch) Prospective Researcher Fellowship PBZHP2-147259 and by a Marie Curie Intra European Fellowship (iCub Body Schema 625727) within the 7th European Community Framework Programme (http://cordis.europa.eu). ZS was supported by the project No. SGS13/203/OHK3/3T/13 of the Czech Technical University in Prague (https://www.cvut.cz/en). IF was supported by the Slovak Grant Agency for Science (VEGA) of the Ministry of Education, Science, Research and Sport of the Slovak Republic (https://www.minedu.sk) and of Slovak Academy of Sciences (SAS, www.sav.sk), project 1/0898/14. GM was supported by the 7th European Community Framework Programme project WYSIWYD (FP7-ICT-612139).

## References

[1] A. S. Leyton and C. S. Sherrington, "Observations on the excitable cortex of the chimpanzee, orangutan, and gorilla," *Quarterly Journal of Experimental Physiology*, vol. 11, no. 2, pp. 135–222, 1917.

[2] W. Penfield and E. Boldrey, "Somatic motor and sensory representation in the cerebral cortex of man as studied by electrical stimulation," *Brain*, vol. 37, pp. 389–443, 1937.

[3] W. Penfield and T. Rasmussen, *The Cerebral Cortex of Man: a Clinical Study of Localization of Function*. Macmillan, 1950.

[4] R. Nelson, M. Sur, D. Felleman, and J. Kaas, "Representations of the body surface in postcentral parietal cortex of macaca fascicularis," *Journal of Comparative Neurology*, vol. 192, no. 4, pp. 611–643, 1980.

[5] "Central processing: The sensory homunculus (fig. 5)," OpenStax College, OpenStax CNX, June 2013. [Online]. Available: http://cnx.org/contents/29cade27-ba23-4f4a-8cbd-128e72420f31@5/Central-Processing

[6] M. C. Crair, "Neuronal activity during development: permissive or instructive?" *Current Opinion in Neurobiology*, vol. 9, no. 1, pp. 88–93, 1999.

[7] P. Vanderhaeghen, Q. Lu, N. Prakash, J. Frisén, C. A. Walsh, R. D. Frostig, and J. G. Flanagan, "A mapping label required for normal scale of body representation in the cortex," *Nature Neuroscience*, vol. 3, no. 4, pp. 358–365, 2000.

[8] M. Granmo, P. Petersson, and J. Schouenborg, "Action-based body maps in the spinal cord emerge from a transitory floating organization," *Journal of Neuroscience*, vol. 28, no. 21, pp. 5494–5503, 2008.

[9] J. H. Kaas and K. C. Catania, "How do features of sensory representations develop?" *Bioessays*, vol. 24, no. 4, pp. 334–343, 2002.

[10] C. von der Malsburg and D. Willshaw, "How to label nerve cells so that they can interconnect in an ordered fashion," *Proceedings of the National Academy of Sciences*, vol. 74, no. 11, pp. 5176–5178, 1977.

[11] J. C. Pearson, L. H. Finkel, and G. M. Edelman, "Plasticity in the organization of adult cerebral cortical maps: a computer simulation based on neuronal group selection," *Journal of Neuroscience*, vol. 7, no. 12, pp. 4209–4223, 1987.

[12] G. I. Detorakis and N. P. Rougier, "A neural field model of the somatosensory cortex: Formation, maintenance and reorganization of ordered topographic maps," *PloS one*, vol. 7, no. 7, p. e40257, 2012.

[13] K. Obermayer, H. Ritter, and K. Schulten, "Large-scale simulation of a self-organizing neural network: Formation of a somatotopic map," *Parallel Processing in Neural Systems and Computers*, pp. 71–74, 1990.

[14] H. Ritter, T. Martinetz, K. Schulten, D. Barsky, M. Tesch, and R. Kates, *Neural computation and self-organizing maps: an introduction*. Addison Wesley Longman Publishing Co., Inc., 1992, ch. Modeling the somatotopic map, pp. 101–117.

[15] T. Stafford and S. P. Wilson, "Self-organisation can generate the discontinuities in the somatosensory map," *Neurocomputing*, vol. 70, no. 10, pp. 1932–1937, 2007.

[16] A. Pitti, Y. Kuniyoshi, M. Quoy, and P. Gaussier, "Modeling the minimal newborn's intersubjective mind: the visuotopic-somatotopic alignment hypothesis in the superior colliculus," *PloS ONE*, vol. 8, no. 7, p. e69474, 2013.

[17] Y. Yamada, H. Kanazawa, S. Iwasaki, Y. Tsukahara, O. Iwata, S. Yamada, and Y. Kuniyoshi, "An embodied brain model of the human foetus," *Scientific Reports*, vol. 6, 2016.

[18] H. Mori and Y. Kuniyoshi, "A human fetus development simulation: Self-organization of behaviors through tactile sensation," in *IEEE 9th International Conference on Development and Learning (ICDL)*. IEEE, 2010, pp. 82–87.

[19] R. Sasaki, Y. Yamada, Y. Tsukahara, and Y. Kuniyoshi, "Tactile stimuli from amniotic fluid guides the development of somatosensory cortex with hierarchical structure using human fetus simulation," in *IEEE 3rd Joint International Conference on Development and Learning and Epigenetic Robotics (ICDL)*, Aug 2013, pp. 1–6.

[20] C. Bartolozzi, L. Natale, F. Nori, and G. Metta, "Robots with a sense of touch," *Nature Materials*, vol. 15, no. 9, pp. 921–925, 2016.

[21] R. S. Dahiya and M. Valle, *Robotic Tactile Sensing*. Springer, 2013.

[22] F. Mastrogiovanni, L. Natale, G. Cannata, and G. Metta, "Special issue on advances in tactile sensing and tactile-based human–robot interaction," *Robotics and Autonomous Systems*, vol. 63, pp. 227–229, 2015.

[23] P. Mittendorfer, E. Yoshida, and G. Cheng, "Realizing whole-body tactile interactions with a self-organizing, multi-modal artificial skin on a humanoid robot," *Advanced Robotics*, vol. 29, no. 1, pp. 51–67, 2015.

[24] B. C.-K. Tee, A. Chortos, A. Berndt, A. K. Nguyen, A. Tom, A. McGuire, Z. C. Lin, K. Tien, W.-G. Bae, H. Wang, P. Mei, H.-H. Chou, B. Cui, K. Deisseroth, T. N. Ng, and Z. Bao, "A skin-inspired organic digital mechanoreceptor," *Science*, vol. 350, no. 6258, pp. 313–316, 2015.

[25] B. D. Argall and A. G. Billard, "A survey of tactile human–robot interactions," *Robotics and Autonomous Systems*, vol. 58, no. 10, pp. 1159–1176, 2010.

[26] U. Martinez-Hernandez and T. Prescott, "Expressive touch: control of robot emotional expression by touch," in *Proc. IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, 2016.

[27] S. Denei, F. Mastrogiovanni, and G. Cannata, "Towards the creation of tactile maps for robots and their use in robot contact motion control," *Robotics and Autonomous Systems*, vol. 63, pp. 293–308, 2015.

[28] G. Cannata, S. Denei, and F. Mastrogiovanni, "Towards automated self-calibration of robot skin," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2010, pp. 4849–4854.

[29] A. Del Prete, S. Denei, L. Natale, F. M., F. Nori, G. Cannata, and G. Metta, "Skin spatial calibration using force/torque measurements," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2011, pp. 3694 –3700.

[30] S. McGregor, D. Polani, and K. Dautenhahn, “Generation of tactile maps for artificial skin,” *PloS one*, vol. 6, no. 11, p. e26561, 2011.

[31] G. Pugach, A. Pitti, and P. Gaussier, “Neural learning of the topographic tactile sensory information of an artificial skin through a self-organizing map,” *Advanced Robotics*, pp. 1–17, 2015.

[32] R. Pfeifer and J. C. Bongard, *How the body shapes the way we think: a new view of intelligence*. Cambridge, MA: MIT Press, 2007.

[33] M. Hoffmann, H. Marques, A. Hernandez Arieta, H. Sumioka, M. Lungarella, and R. Pfeifer, “Body schema in robotics: A review,” *Autonomous Mental Development, IEEE Transactions on*, vol. 2, no. 4, pp. 304–324, Dec 2010.

[34] A. Roncone, M. Hoffmann, U. Pattacini, L. Fadiga, and G. Metta, “Peripersonal space and margin of safety around the body: learning tactile-visual associations in a humanoid robot with artificial skin,” *PLoS ONE*, vol. 11, no. 10, p. e0163713, 2016.

[35] M. Hoffmann and N. Bednarova, “The encoding of proprioceptive inputs in the brain: knowns and unknowns from a robotic perspective,” in *Kognice a umely zivot XVI [Cognition and Artificial Life XVI]*, M. Vavrecka, O. Becev, M. Hoffmann, and K. Stepanova, Eds., 2016, pp. 55–66.

[36] G. Metta, L. Natale, F. Nori, G. Sandini, D. Vernon, L. Fadiga, C. von Hofsten, K. Rosander, M. Lopes, J. Santos-Victor, A. Bernardino, and L. Montesano, “The iCub humanoid robot: An open-systems platform for research in cognitive development,” *Neural Networks*, vol. 23, no. 8-9, pp. 1125–1134, 2010.

[37] A. Parmiggiani, M. Maggiali, L. Natale, F. Nori, A. Schmitz, N. Tsagarakis, J. S. Victor, F. Becchi, G. Sandini, and G. Metta, “The design of the iCub humanoid robot,” *International Journal of Humanoid Robotics*, vol. 9, no. 04, 2012.

[38] P. Maiolino, M. Maggiali, G. Cannata, G. Metta, and L. Natale, “A flexible and robust large scale capacitive tactile system for robots,” *IEEE Sensors Journal*, vol. 13, no. 10, pp. 3910–3917, 2013.

[39] M. Hoffmann, Z. Straka, I. Farkas, M. Vavrecka, and G. Metta, “Supporting materials.” [Online]. Available: https://github.com/matejhof/robotic-homunculus-supporting-materials

[40] T. Kohonen, “Self-organized formation of topologically correct feature maps,” *Biological Cybernetics*, vol. 43, no. 1, pp. 59–69, 1982.

[41] ——, “The self-organizing map,” *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1464–1480, 1990.

[42] E. Jones, “Cortical and subcortical contributions to activity-dependent plasticity in primate somatosensory cortex,” *Annual Review of Neuroscience*, vol. 23, no. 1, pp. 1–37, 2000.

[43] J. Vesanto, J. Himberg, E. Alhoniemi, and J. Parhankangas, “Self-organizing map in Matlab: the SOM toolbox,” in *Proceedings of the Matlab DSP Conference*, vol. 99, 1999, pp. 16–17.

[44] E. Alhoniemi, J. Himberg, J. Parhankangas, and J. Vesanto, “Som toolbox-online documentation,” 2003. [Online]. Available: http://www.cis.hut.fi/projects/somtoolbox/package/docs2/somtoolbox.html

[45] D. Polani, “Measures for the organization of self-organizing maps,” in *Self-Organizing Neural Networks*, ser. Studies in Fuzziness and Soft Computing, U. Seiffert and L. Jain, Eds. Springer, 2002, vol. 78, pp. 13–44.

[46] H.-U. Bauer and K. R. Pawelzik, “Quantifying the neighborhood preservation of self-organizing feature maps,” *IEEE Transactions on Neural Networks*, vol. 3, no. 4, pp. 570–579, 1992.

[47] G. H. Büscher, R. Kõiva, C. Schürmann, R. Haschke, and H. J. Ritter, “Flexible and stretchable fabric-based tactile sensor,” *Robotics and Autonomous Systems*, vol. 63, pp. 244–252, 2015.

[48] P. Mittendorfer and G. Cheng, “Humanoid multimodal tactile-sensing modules,” *IEEE Transactions on Robotics*, vol. 27, no. 3, pp. 401–410, 2011.

[49] ——, “3d surface reconstruction for robotic body parts with artificial skins,” in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012.

[50] E. Kandel, J. Schwartz, and T. Jessell, *Principles of Neural Science*. McGraw-Hill, 2000, vol. 4.

[51] G. J. Gerling, I. I. Rivest, D. R. Lesniak, J. R. Scanlon, and L. Wan, “Validating a population model of tactile mechanotransduction of slowly adapting type i afferents at levels of skin mechanics, single-unit response and psychophysics,” *Haptics, IEEE Transactions on*, vol. 7, no. 2, pp. 216–228, 2014.

[52] A. Chortos, J. Liu, and Z. Bao, “Pursuing prosthetic electronic skin,” *Nature Materials*, vol. 15, no. 9, pp. 937–950, 2016.

[53] H. P. Saal and S. J. Bensmaia, “Touch is a team effort: interplay of submodalities in cutaneous sensibility,” *Trends in neurosciences*, vol. 37, no. 12, pp. 689–697, 2014.

[54] S. S. Hsiao and F. Vega-Bermudez, “Attention in the somatosensory system,” in *The somatosensory system: Deciphering the brain’s own body image*, R. J. Nelson, Ed. CRC Press, 2001, pp. 197–218.

[55] J. Xing and G. Gerstein, “Networks with lateral connectivity. iii. plasticity and reorganization of somatosensory cortex,” *Journal of Neurophysiology*, vol. 75, no. 1, pp. 217–232, 1996.

[56] K. Obermayer, H. Ritter, and K. Schulten, “A principle for the formation of the spatial structure of cortical feature maps,” *Proceedings of the National Academy of Sciences, USA*, vol. 87, pp. 8345–8349, 1990.

[57] J. Sirosh and R. Miikkulainen, “Cooperative self-organization of afferent and lateral connections in cortical maps,” *Biological Cybernetics*, vol. 71, pp. 66–78, 1994.

[58] J.-L. R. Stevens, J. Law, J. Antolik, and J. Bednar, “Mechanisms for stable, robust, and adaptive development of orientation maps in the primary visual cortex,” *Journal of Neuroscience*, vol. 33, pp. 15 747–15 766, 2013.

[59] N. P. Rougier and Y. Boniface, “Dynamic self-organising map,” *Neurocomputing*, vol. 74, pp. 1840–1847, 2011.

[60] A. Roncone, M. Hoffmann, U. Pattacini, and G. Metta, “Automatic kinematic chain calibration using artificial skin: self-touch in the icub humanoid robot,” in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, 2014, pp. 2305–2312.

**Matěj Hoffmann** received his Mgr. (M.Sc.) degree in Computer Science, Artificial Intelligence at Faculty of Mathematics and Physics, Charles University in Prague, Czech Republic, in 2006. Between 2006 and 2013 he completed his PhD degree and then served as senior research associate at the Artificial Intelligence Laboratory, University of Zurich, Switzerland (Prof. Rolf Pfeifer). From May 2013 he worked at the iCub Facility of the Italian Institute of Technology with Prof. Giorgio Metta, between 2014 and 2016 as a Marie Curie Experienced Researcher Fellow. In 2017 he joined Dept. Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague. His main research interest is embodied cognition, in particular the mechanisms underlying body representations and sensorimotor contingencies in humans and their implications for increasing the autonomy, resilience and robustness of robots.

**Zdeněk Straka** received his Bc. degree (with Honors) in Robotics and masters (Ing.; with Honors) degree in Artificial Intelligence from the Faculty of Electrical Engineering, Czech Technical University in Prague in 2014 and 2016 respectively. His Bc. thesis on the development of tactile maps in a humanoid robot was awarded the Dean’s prize. From September 2016 he is a PhD student at the Center for Machine Perception, CTU in Prague. His research interests include neurorobotics, neural networks, and machine learning. He is particularly interested in applying machine learning methods to body and peripersonal space representations of humanoid robots.

**Igor Farkaš** received the masters degree ”Ing.” (with Honors) in technical cybernetics in 1991, and the Ph.D. degree in applied informatics in 1995, both from the Slovak University in Technology in Bratislava, Slovakia. In 1998 he was a Fulbright fellow at the Department of Computer Science, University of Texas at Austin, USA. From 2000 to 2003 he was a postdoctoral fellow at the Department of Psychology, University of Richmond, VA, USA. In 2005 he was Humboldt fellow at the Department of Computational Linguistics and Phonetics, Saarland University in Saarbrücken, Germany. In 2014 he became full professor of informatics at the Faculty of Mathematics, Physics and Informatics, Comenius University in Bratislava, Slovak Republic. His research interests include models of artificial neural networks and their applications in cognitive modeling, mainly natural language and robotics. Since 2013, he has been a member of the IEEE Computational Intelligence Society Neural Network Technical Committee.

**Michal Vavrečka** works at CTU in Prague. He focuses on knowledge representation, namely the development of multimodal representations. Michal developed multimodal architectures for grounding symbols in the area of spatial navigation to represent static (up, down etc.) and dynamic (around, through etc.) spatial prepositions. The main goal is to test the methods of unsupervised learning in the process of knowledge acquisition in terms of multimodal integration. The second branch of his research is focused on cognitive neuroscience. He is interested in neural correlates of spatial navigation especially the localization of brain structures involved in egocentric and allocentric frames of reference processing.

**Giorgio Metta** is Vice Scientific Director at the Istituto Italiano di Tecnologia (IIT) and Director of the iCub Facility Department at the same institute. He coordinates the development of the iCub robotic project. He holds an MSc cum laude (1994) and PhD (2000) in electronic engineering both from the University of Genoa. From 2001 to 2002, he was postdoctoral associate at the MIT AI-Lab. He was previously with the University of Genoa and since 2012 Professor of Cognitive Robotics at the University of Plymouth (UK). He is also deputy director of IIT delegate to the training of young researchers. He is member of the board of directors of euRobotics aisbl, the European reference organization for robotics research. Giorgio Metta research activities are in the fields of biologically motivated and humanoid robotics and, in particular, in developing humanoid robots that can adapt and learn from experience. He has been working as principal investigator and research scientist in about a dozen international as well as national funded projects.

# Appendix G

# Peripersonal space and margin of safety around the body: learning tactile-visual associations in a humanoid robot with artificial skin

Roncone, A., Hoffmann, M., Pattacini, U., Fadiga, L. and Metta, G. (2016). Peripersonal space and margin of safety around the body: learning tactile-visual associations in a humanoid robot with artificial skin. *PLoS ONE* 11 (10): e0163713.

Open Access. DOI: `http://dx.doi.org/10.1371/journal.pone.0163713`

Author contributions: The contribution of M. Hoffmann was 35%.

PLOS | ONE

RESEARCH ARTICLE

# Peripersonal Space and Margin of Safety around the Body: Learning Visuo-Tactile Associations in a Humanoid Robot with Artificial Skin

**Alessandro Roncone**[1,2], **Matej Hoffmann**[1]*, **Ugo Pattacini**[1], **Luciano Fadiga**[3,4], **Giorgio Metta**[1]

**1** iCub Facility, Istituto Italiano di Tecnologia, Genova, Italy, **2** Social Robotics Lab, Computer Science Department, Yale University, New Haven, CT, United States of America, **3** Robotics, Brain, and Cognitive Sciences Department, Istituto Italiano di Tecnologia, Genova, Italy, **4** Section of Human Physiology, Ferrara University, Ferrara, Italy

* matej.hoffmann@iit.it

**Citation:** Roncone A, Hoffmann M, Pattacini U, Fadiga L, Metta G (2016) Peripersonal Space and Margin of Safety around the Body: Learning Visuo-Tactile Associations in a Humanoid Robot with Artificial Skin. PLoS ONE 11(10): e0163713. doi:10.1371/journal.pone.0163713

**Editor:** Mikhail A. Lebedev, Duke University, UNITED STATES

**Received:** February 16, 2016

**Accepted:** August 17, 2016

**Published:** October 6, 2016

**Data Availability Statement:** All relevant data and scripts to generate figures are in this public repository: https://github.com/alecive/peripersonal-space-margin-of-safety-data.

**Funding:** AR was supported by 7th European Community Framework Programme (http://cordis.europa.eu) Xperience (FP7-ICT-270273). MH was supported by the Swiss National Science Foundation (www.snf.ch) Prospective Researcher Fellowship PBZHP2-147259 and by a Marie Curie Intra European Fellowship (iCub Body Schema 625727) within the 7th European Community

## Abstract

This paper investigates a biologically motivated model of peripersonal space through its implementation on a humanoid robot. Guided by the present understanding of the neurophysiology of the fronto-parietal system, we developed a computational model inspired by the receptive fields of polymodal neurons identified, for example, in brain areas F4 and VIP. The experiments on the iCub humanoid robot show that the peripersonal space representation i) can be learned efficiently and in real-time via a simple interaction with the robot, ii) can lead to the generation of behaviors like avoidance and reaching, and iii) can contribute to the understanding the biological principle of motor equivalence. More specifically, with respect to i) the present model contributes to hypothesizing a learning mechanisms for peripersonal space. In relation to point ii) we show how a relatively simple controller can exploit the learned receptive fields to generate either avoidance or reaching of an incoming stimulus and for iii) we show how the robot can select arbitrary body parts as the controlled *end-point* of an avoidance or reaching movement.

## Introduction

The peripersonal space (PPS) is of special relevance for the life of any complex animal. When objects enter the peripersonal space, they can be reached for, grasped, or be a threat, evoking for example an avoidance response. Peripersonal space thus deserves special attention and probably justifies the specific neural circuitry devoted to its representation. The brain has to dynamically integrate information coming from several modalities: motoric, visual, auditory or somatosensory. In primates, the evidence derived primarily from recordings in the macaque identifies a specific fronto-parietal network of neurons as the circuitry responsible for the

Framework Programme. LF was supported by the 7th European Community Framework Programme project POETICON++ (FP7-ICT-2011-7) and by the MIUR-PRIN (http://prin.miur.it/) 2010MEFNF7_003 grant. UP and GM were supported by the 7th European Community Framework Programme project WYSIWYD (FP7-ICT-612139).

**Competing Interests:** The authors have declared that no competing interests exist.

representation of peripersonal space, as well as the connection to extant behavior (e.g., [1–3]). In the frontal lobe, the principal convergence locus has been discovered to be area F4 of the ventral premotor cortex [4–6] including the region of the spur of the arcuate sulcus [7]. In the parietal lobe, the area most strongly connected to area F4 is area VIP (Ventral Intraparietal [8]). In spite of the fact that observations report the presence of auditory responses [9], in this work we leave audition aside and focus instead on the integration of visual and tactile inputs.

A large part of peripersonal space coding can presumably be attributed to populations of polymodal neurons that, in addition to motor discharge, have tactile and visual receptive fields (RFs). Visual RFs usually extend from the tactile ones in the space around the respective body segment (see e.g. [4, 5]; for a review, see [1–3]). Furthermore, the visual RFs are often coded in the same frame of reference (FoR) of the corresponding body part and, therefore, during active or passive mobilization, they move with the body part in 3D space. This suggests that motor and proprioceptive information is integrated in a body-part-centered encoding. A good part of the evidence coming from the monkey is presumably informative in the case of humans as well [10].

Timely and appropriate object-directed actions in the peripersonal space are crucial for the survival of the animal. Depending on the context, actions may constitute either an approaching or an avoidance behavior. In the case of avoidance behavior, this creates a "margin of safety" around the body, such as the flight zone of grazing animals or the multimodal attentional space that surrounds the skin in humans [2]. An analogous behavior is desirable in general-purpose robots as well, when significant interaction is expected to happen in unconstrained environments. However, to date, robot controllers largely concentrate on the *end-point* as the only part that enters in physical contact with the environment. The rest of the body is typically represented as a kinematic chain, the volume and surface of the body itself rarely taken into account. Sensing is dominated by "distal" sensors, like cameras, whereas the body surface is "numb". As a consequence, reaching in cluttered, unstructured environments poses severe problems, as the robot is largely unaware of the full occupancy of its body, limiting the safety of the robot and the surrounding environment. This is one of the bottlenecks that prevent robots from working alongside human partners.

While individual components that presumably constitute the representations of space around the body can be studied in isolation using computational models in simplified (for example 2-dimensional) scenarios, their interactions are difficult to model without an articulated body with corresponding sensorimotor capacities and actual interaction with the environment. Indeed, in animals and humans, these representations are gradually formed through physical interaction with the environment and in a complex interplay of body growth and neural maturation processes. Self-touch (also called double-touch) is presumably one of the behaviors that impact the formation of multimodal body representations. For example, "*by 2-3 months, infants engage in exploration of their own body as it moves and acts in the environment. They babble and touch their own body, attracted and actively involved in investigating the rich intermodal redundancies, temporal contingencies, and spatial congruence of self-perception*" [11]. Such behaviors may initially be reflexive and controlled by spinal circuitry—the wiping/scratch reflex has been demonstrated in frogs [12, 13], though its existence is debated in humans [14]—but progressively become more complex and voluntary. These contingencies and congruences that arise occur across different motor and sensory modalities, with the motor/proprioceptive and tactile starting already in prenatal stage. Vision is presumably incorporated later, during the first months after birth hand in hand with the maturation of the visual system (see e.g., [15]). Perhaps even later, contingencies will encompass external objects (this loosely resembles the sensorimotor stage of development put forth by Piaget, e.g. [16]).

This simplified developmental timeline constitutes the skeleton of our work in the humanoid robot. The humanoid in question is the iCub, a child robot designed to support studies on artificial cognitive systems [25]. The iCub has a human-like morphology and a subset of the sensory capacities of the human body. Lately, it has been equipped with a set of tactile sensors [26], which provide information about local pressure upon contact with an object or, generically, any part of the environment. We are concretely in the position of studying how motor-proprioceptive-tactile and visuo-tactile associations are developed via an artificial learning process. The robot can and will therefore establish a margin of safety by interacting with its own body and the environment, extending its cutaneous tactile surface into the 3D space surrounding it. An overview of the developmental timeline is provided in Tables 1–4, with putative developmental milestones in the left column and their robotic counterparts on the right.

The robotics implementation departs in many respects from the mechanisms that presumably operate in the primate brain. The correspondence between biology and robotics is often established at a behavioral level rather than in the details of the implementation. In particular, for mostly practical reasons, we assume that the robot's kinematics and mapping of tactile information into reference frames is given. The implementation of the double-touch behavior itself (from [18]) is taken as a primitive. Conversely, learning/calibration of the spatial receptive fields around individual taxels (tactile elements) is primarily addressed here and relates to biology.

Building on the developmental pathway outlined above, we model peripersonal space on a humanoid robot equipped with full-body tactile sensors. Our model keeps in register each "spatial" visual RF to a taxel of the robot's skin. Starting from an initial "blank slate", the distance and velocity of a stimulus entering any given RF is recorded, together with information on whether the object had eventually contacted the selected tactile element. Distance and velocity of the stimulus are measured with respect to each taxel, in real time and in parallel. In this model, RFs are proxies for the neural responses, each of them represented by a probability density function. Probabilities are updated incrementally and carry information about the likelihood of a particular stimulus (e.g. an object approaching the body) eventually contacting the specific taxel at hand. We use the distance to the taxel and its time to contact (distance/velocity) to compactly identify the stimulus in a bi-dimensional parameter space.

**Table 1. Developmental milestone 1: *"Bare" or "blind" double-touch.***

| | Developmental Milestone | Robotics Implementation |
|---|---|---|
| 1a. | **Double-touch from body babbling or mediated by reflexes**. Fetuses as well as infants spontaneously contact their bodies, giving rise to self-touch events. Correspondence between *motor* (how to command a limb to touch a specific body part) and *tactile* (cutaneous stimulation on touching and touched body part) information are established [11, 17]. | **Double-touch using inverse kinematics**. In the iCub robot, we used a solution for double-touch developed in Roncone et al. [18]. This capitalizes on an existing kinematic model of the robot as well as calibration of the artificial skin with respect to a common FoR and employs a modified inverse kinematics solver. This solution automatically encompasses different arm configurations, since current joint positions automatically enter the kinematic representation. *Subject to learning*: As one arm approaches and eventually contacts another body part (the contralateral arm in our case), the position and velocity of the approaching arm are acquired from current joint angle values and kinematic model of the robot and then remapped into the FoR of the taxels on the touched limb. These taxels then, in parallel and in their individual FoRs, learn a probabilistic representation of the likelihood of a stimulus—the approaching limb in this case—contacting them. |
| 1b. | **Invariance with respect to the configuration of the "touched" limb**. If movement is directed to a body part that can assume different configurations with respect to the body frame (e.g. arm), the information about the current position of this body part needs to be taken into account—presumably using *proprioceptive information*. Some form of remapping of tactile information into external (to the skin) reference frames seems necessary (see Heed et al. [19] for a review). *Outcome*: Prediction of double-touch from motor/proprioceptive information. | |

doi:10.1371/journal.pone.0163713.t001

**Table 2. Developmental milestone 2:** ***Double-touch with vision*****.**

| | Developmental Milestone | Robotics Implementation |
|---|---|---|
| **2a.** | **The motor/proprioceptive information about the position of the approaching arm is augmented by vision**. Here we assume that its 3D position with respect to a certain reference frame can be retrieved (using stereopsis). This can then be used to develop *visuo-tactile associations* able to predict incoming contact based on visual information. | **Visual tracking with extraction of 3D coordinates; head and eye kinematics**. A visual tracker (specified in the following sections) is used to detect and extract coordinates of approaching limb into a body-centered reference frame. A model of eye and head kinematics together with current joint values are used to perform the necessary kinematic transformations. Further, a gaze controller [20] is employed to track the approaching stimulus (fingertip in this case). Different limb as well as head and eye configurations are automatically taken into account. *Subject to learning*: Probabilistic representation of stimuli eventually resulting in double-touch, but this time utilizing visual information about the approaching limb. |
| **2b.** | **Invariance with respect to the configuration of the "touched" limb**. Similarly to **1b**, if the "touched" body part can assume different configurations, this needs to be taken into account in order to register the visuo-tactile association correctly—presumably in a reference frame centered on the body part. Again, *proprioceptive* signals about limb configuration can provide this information. | |
| **2c.** | **Invariance with respect to the head and eye configuration**. The touching limb needs to be followed in space by gaze. For correct registration of the active limb's position and subsequent coordinate transformations, *proprioceptive* information about the current neck and eyes configurations is needed (see [21, 22] regarding the role of gaze in reaching to somatosensory targets). *Outcome*: Prediction of double-touch while extracting the position and velocity of the approaching arm from visual information. | |

doi:10.1371/journal.pone.0163713.t002

For learning probabilities, we explore two stimulation modalities: i) self-touch and, more generically, ii) objects moving toward the body surface. In the first case, the stimulus is generated autonomously by the robot—for example, a finger touching the contralateral arm. The robot executes self-touching behaviors and uses proprioceptive signals to measure the approach kinematics, which in turn constitute the training set to estimate probability densities. In the second modality, stimuli are generated by any object in the vicinity of the body surface and perceived visually and through its contact with the skin.

**Table 3. Developmental milestone 3:** ***Visuo-tactile associations pertaining to external objects*****.**

| | Developmental Milestone | Robotics Implementation |
|---|---|---|
| **3a.** | **Tactile-visual-proprioceptive learning from any approaching stimulus**. The tactile-visual-proprioceptive association learned in previous stage is generalized and applied to any objects nearing the skin. Visual perception of own approaching body parts is substituted by detection and tracking of moving objects in the body surroundings. We did not consider further stimuli approaching the face (e.g., [23]), where expanding optic flow fields may in fact be at use (e.g., [24]), but objects nearing the limbs. *Outcome*: Prediction of contact of skin parts—with own body or with generic objects—using visual information. | Same as Milestone **2** above, but using a different visual perception pipeline able to accommodate arbitrary objects (detailed in the paper). *Subject to learning*: Probabilistic representation of stimuli eventually resulting in contact with the skin, utilizing visual information about approaching objects. |

doi:10.1371/journal.pone.0163713.t003

**Table 4. Developmental milestone 4: *Exploitation of learned associations*.**

| | Developmental Milestone | Robotics Implementation |
|---|---|---|
| **4a.** | **Avoidance behaviors**. Prediction of contact is exploited to trigger coordinated avoidance behaviors w.r.t. either the own body (i.e. avoiding self-collisions) or the external world (i.e. avoiding incoming potentially harmful objects) (e.g., [2]). *Outcome*: Effective "safety margin around the body". | **Distributed avoidance / catching controller**. Taxels with activation above a certain threshold contribute to a resulting movement vector that is executed by a Cartesian controller. The avoidance / reaching differs only in the direction of the final movement vector. |
| **4b.** | **"Reaching" with arbitrary body parts behaviors**. The peripersonal space representation facilitates actions toward nearby objects, allowing to reach for them with any body part. *Outcome*: reaching actions with arbitrary body parts. | |

doi:10.1371/journal.pone.0163713.t004

There are a number of computational models addressing phenomena related to peripersonal space representations. A major component of many of them is coordinate transformations, which seem inevitable in order to code visual information in body-part centered FoRs; this has been investigated extensively and several connectionist models have been proposed (e.g., [23, 27, 28]). On the other hand, Magosso et al. [29] took FoR transformations for granted and focused on the mechanisms of tactile and visual interaction. They proposed a neural network that models unimodal (visual and tactile) and bimodal representations of an imaginary left and right body part and demonstrated a number of phenomena reported in humans (e.g. tactile extinction). Some of the studies targeting body schema and peripersonal space representation models were reviewed in Hoffmann et al. [30]. Since platforms with tactile sensing are rare, most of the work has focused on the interaction of visual and proprioceptive information (in robotics typically equated with joint angles from encoders). For example, Antonelli et al. [31, 32] developed models in different humanoid robots, focusing mainly on peripersonal space in the sense of space within reach and the visual aspects thereof. A number of embodied models were also developed by Asada and colleagues. Hikita et al. [33] used a humanoid robot and employed a bio-inspired architecture (self-organizing maps, Hebbian learning, and attention module) to learn the visual receptive field around the robot's hand and its extension when using a tool—inspired by the behavior of the "distal" type neurons reported by Iriki et al. [34]. Touch was only emulated and used to trigger the visuo-proprioceptive association. Finally, most related to our approach, Fuke et al. [35] used a simulated robot touching itself on the face to model the putative mechanism leading to the visual and tactile response properties of neurons in the ventral intraparietal area (VIP). A hierarchical architecture with visual, proprioceptive and tactile modality was used. After learning, as the robot's hand approached its face, contact with the skin could be predicted.

In robotics, safe interaction, especially when involving humans, is a crucial need of future assistive machines. There is necessity for technologies that allow robots to acquire some form of "whole-body" and "nearby-space" awareness. Traditionally, a significant body of work has been produced in the context of obstacle-avoidance planners, able to compute safe end-effector trajectories off-line if provided with complete knowledge of a static environment and a precise kinematic model. These approaches fall short in presence of modeling errors or when environments change dynamically. To this end, the classic planning techniques had to be complemented by reactive strategies such as the potential field approach [36]. More recently, frameworks taking the whole occupancy of a robot body into account have appeared: Flacco et al. [37] proposed a motion controller with online collision avoidance for both end-effector and the manipulator body; Frank et al. [38] proposed a modular framework (MoBeE) where a planner can be overridden by a reactive controller. Still, the performance of systems relying on distal sensing (such as from cameras or depth sensors) degrades if the perception of the environment is not reliable or the model of the robot kinematics inaccurate. A feedback loop that is as close as possible to the interaction itself is needed.

In recent years, tactile systems have been proposed as a way to close the loop precisely where the interaction occurs. However, the lack of suitable platforms limits research in this direction: although diverse tactile sensing technologies have been developed (see [39] for a review), robots with whole-body tactile sensing have been mostly unavailable. Alternative solutions relied on force/torque sensing and impedance control schemes that ensure compliant behavior of the platform on contact (e.g., [40]). Shimizu et al. [41] used force/torque feedback together with encoder information to develop self-protective reflexes and global reactions for the iCub robot. Distributed sensing over the whole surface of a robotic manipulator was used by Mittendorfer and Cheng [42]. Utilizing information from accelerometers from their multimodal "skin" during a motor exploration phase, the direction of movement of every sensory unit in response to every motor could be learned. Activations of infra-red distance sensors on the same sensory unit could then be used to trigger local avoidance reflexes to approaching objects. Finally, Jain et al. [43] devised a controller that allows for reaching in clutter while taking into account multiple contacts and keeping the forces within set limits. The solution was verified on a robot featuring a tactile-sensitive forearm. However, solutions combining interaction-based and contact-less (distal sensing) approaches are rare ([44] being a notable exception). This is where our work exploiting visual and whole-body tactile information ties in.

In this work, we set forth to implement a model of peripersonal space that includes self-tuning abilities in the form of learning from examples. Specifically, we do not model the acquisition of the FoR transformations but rather we focus only on the construction of the responses of the RFs. We build on our previous work [45], where we presented a simplified version of the model dealing solely with approaching external objects and registering their distance. Here we extend this work by presenting a complete developmental timeline, in which examples are first collected through self-exploration or self-touch, resulting in concurrent motor-tactile and visuo-tactile stimulation of different areas of the body. This is then complemented by external approaching objects. Furthermore, the RFs' representations take into account the time to contact of the incoming stimulus. Finally, the acquired RFs are used in a controller to implement avoidance and reaching behaviors thus implicitly testing their performance.

This article is structured as follows. In the Results section, the properties of the proposed model are first verified in simulation (Section Learning in a single taxel model) and then on the iCub (Section Learning in the real robot). Finally, the peripersonal space representation is used to generate avoidance as well as "reaching" behavior using arbitrary body parts of the robot (Section Exploitation of the learned associations). This is followed by Discussion and Conclusion, which contains a summary, limitations of the model and future work. A detailed description of the experimental setup and the proposed computational model is presented in Section Materials and Methods.

## Results

Results from four different experimental scenarios are reported (we refer the reader to Tables 1–4 above for an overview of the developmental timeline). First, the behavior of the proposed representation is studied in a simulated single taxel model (Section Learning in a single taxel model). Second, we demonstrate how the robot can learn tactile-motor and tactile-visual representations via a double-touch scenario and by tracking arbitrary objects as they near the skin.

Finally, the utility of the learned representations is demonstrated in the avoidance and reaching scenario that exploits the tactile-visual representations learned previously. The source code has been released online with an open source license and it is readily available for any iCub robot [46]. All the relevant data and scripts needed to reproduce the results shown are accessible at the public repository [47].

## Representation of “Space Around the Body”

We have chosen a distributed representation whereby each taxel learns a collection of probabilities regarding the likelihood of being touched by a moving object. The physiology of the observed neural RFs suggests that their extension in space is modulated by the speed of the incoming stimulus. In addition, the relative position of the stimulus with respect to the receptive field (RF) clearly determines the activation strength of a given neuron. Inspired by these considerations we define a parameter space of two variables: (i) distance from the taxel *D*; (ii) time to contact *TTC*. *TTC* is calculated from the distance *D* and velocity of the incoming stimulus. Fig 1 illustrates the receptive field around one taxel of the forearm and the two main scenario types: self-touch and an external object approaching toward the body. *D* and *TTC* can be calculated in the reference frame of each taxel. Practically, this is possible because of the existing calibration procedure of the robot skin due to del Prete et al. [48] and a full model of the robot’s kinematics derived from CAD data, including the head and eyes [49]—as detailed in the Experimental Setup Section. For stimuli perceived visually, additional processing involving stereo vision is required. In fact, any observation is mapped into the iCub Root FoR (located around its waist) and subsequently transformed to the reference frames of individual taxels. It is important to note that measurements are affected by parametric errors in addition to their intrinsic measurement noise (the modeling errors are discussed in detail in Section Kinematic model and coordinate transformations). The effect of modeling errors can be, for example, that stimuli that are in physical contact with the skin can be perceived as seemingly penetrating the robot surface when employing a sequence of coordinate transformations using the kinematic model and current joint angle measurements. Subsequently, this results in a negative measure of distance *D* with respect to the taxel surface normal. Conversely, if the errors bring about an offset in the opposite direction, an actual contact on the robot’s skin may correspond to a perceived positive distance. Our training data will be affected systematically by these errors which reflect onh the estimated probability densities.

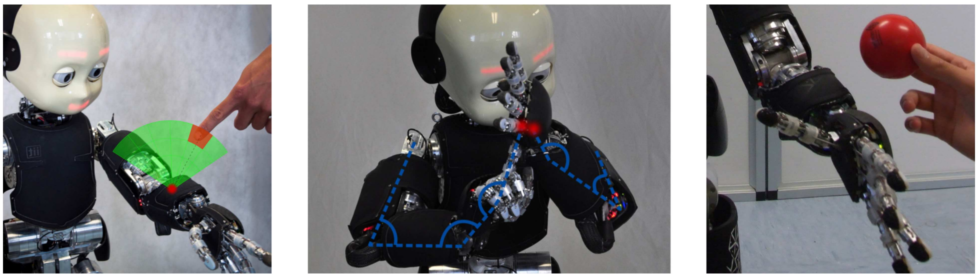


**Fig 1. Illustration of the setup of different scenarios. (left)** Receptive field above one of the left forearm taxels. **(middle)** iCub double-touch behavior with a simplified schematic of the kinematics and joint angles. **(right)** An object approaching the left forearm.

doi:10.1371/journal.pone.0163713.g001

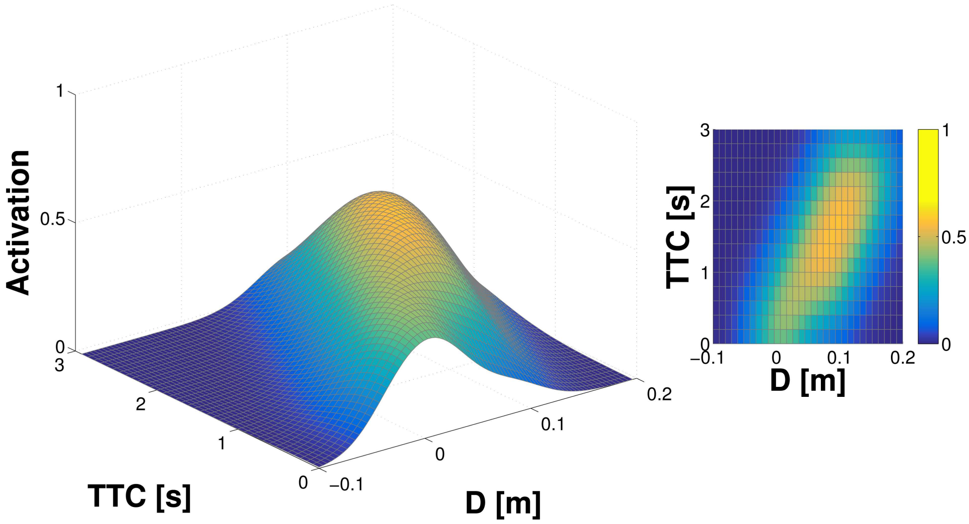


**Fig 2. Representation learned in single taxel model.** *D* and *TTC* estimated from distance and velocity of the object. **(Left)** Full 3D graph of the representation. The *z*−axis is given by the activation—estimate of the probability of object eventually landing on the taxel. **(Right)** 2D projection; third dimension preserved in the color map.

doi:10.1371/journal.pone.0163713.g002

## Learning in a single taxel model

The properties of the learning procedure as well as the proposed representation are investigated in a single taxel model (as specified in Section Monte Carlo simulation of a single taxel). The results from 500 iterations of the simulation—500 objects being "thrown" toward the taxel—are illustrated in Fig 2. They show the representation of the "probability density" (it is not a real probability density—see Section Internal representation) after learning and smoothing using the adapted Parzen window method: the full landscape on the left and its projection in 2D with color coding (the probability of contact) on the right. A clear "ridge" can be seen in both plots, which corresponds to the trajectories of objects as they approach the taxel and both *D* and *TTC* are decreasing. The contact with the taxel occurs at both *D* and *TTC* equal to 0.

In a second simulation, in order to better approximate the experimental conditions encountered by the real robot, two additional features are added to the model. First, Gaussian noise is added to the measurement of position and velocity (and hence *D* and *TTC*). Second, we account for the fact that the object position and velocity measurements in the real robot are subject not only to random, but also to systematic errors. In particular, in both tactile-motor (double-touch) and tactile-visual scenarios, the coordinate transformations needed to map the approaching object to the FoR of individual taxels rely on the model of the robot kinematic structure and its visual system, which are subject to errors (see Section Kinematic model and coordinate transformations). To clearly demonstrate the effect of this on the representation, we introduce a significant systematic offset ($-10cm$) to the simulation. The results for this configuration—noise and systematic error—can be seen in Fig 3. The Gaussian noise results in an overall broader profile of the activation landscape. The offset can be clearly seen in the distance axis, with the "ridge" of high activations cutting the $x$−axis in the negative domain.

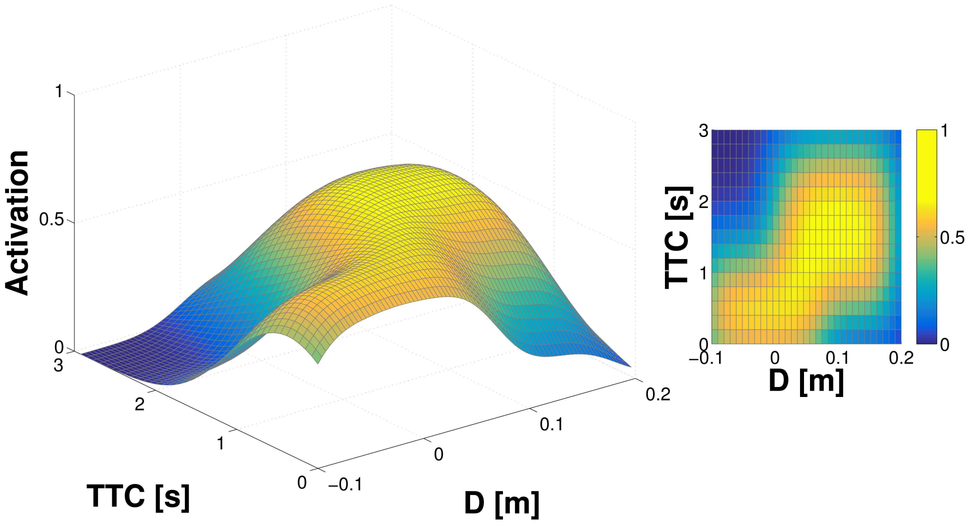


**Fig 3. Representation learned in single taxel model with noise and systematic error (−10*cm* offset).** See text for details.

doi:10.1371/journal.pone.0163713.g003

## Learning in the real robot

The proposed method is then tested in a real-world setup where real, physical stimuli approach the iCub's skin. We investigate the learning of peripersonal space representations in three scenarios corresponding to the putative developmental milestones as discussed in the Introduction. Initially, learning involves exclusively tactile and motor signals (cf. Table 1) as induced by self-touching behaviors (Section Tactile-motor learning: double-touch). In the second phase—Section Tactile-visual learning from double-touch, visual information replaces motor information about the "touching" arm (corresponding to Table 2). Finally, this approach is generalized to any incoming external stimulus that contacts the skin (Table 3) in Section Tactile-visual learning using external objects. An overall comparison of the representations learned in the different scenarios as well as an analysis of the learning process for two adjacent taxels is shown in Section Interim discussion on learning in the real robot.

Table 5 provides a quantitative overview of the data sets collected in the three scenarios. In every case, the skin parts involved are listed, along with the experimental time elapsed, number

**Table 5. Learning in the real robot.** Comparison between three experimental sessions performed on the iCub robot. For each session and each body part under consideration, the elapsed time in minutes (**ET**), the number of trials (**#T**), and the total number of input samples (**#S**) are shown. See text for details.

| Experiment | Body Part | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Left Forearm (internal) | | | Left Forearm (external) | | | Right Hand | | |
| | ET[min] | #T | #S | ET[min] | #T | #S | ET[min] | #T | #S |
| Tactile-motor | 31 | 82 | 3512 | – | – | – | – | – | – |
| Tactile-visual (double-touch) | 30 | 45 | 1166 | – | – | – | – | – | – |
| Tactile-visual (ext. objects) | 23 | 53 | 1886 | 17 | 34 | 1348 | 44 | 77 | 2833 |

doi:10.1371/journal.pone.0163713.t005

of trials (i.e. the number of independent stimuli nearing the robot's skin), and total number of samples (an average of 37 samples per trial were recorded). The movements were directed toward the internal part of the left forearm in all the scenarios—this skin region facilitates the self-touch behavior in the robot—with the same 8 taxels subject to learning. In addition, to demonstrate the generality of the approach, the outer part of the left forearm (4 taxels) as well as the right hand (skin on the palm, 4 taxels) were targeted in the tactile-visual scenario with external objects.

It is worth noting that the data collection and learning process was fast (summing up to 142 minutes for all the experiments reported together). In fact, even a single positive (i.e. touch of the skin) trial gives rise to a usable representation (cf. Section Interim discussion on learning in the real robot below). This is considered a significant merit of the proposed approach, since the algorithm can be used on-line and in real-time without an *a priori* batch learning session: the peripersonal space representations immediately provide prior-to-contact activations and are then refined over time. The smoothing approach used (Parzen window applied to the discrete domain) is specifically responsible for this in the context of undersampled spaces.

## Tactile-motor learning: double-touch

The first experiment on the real robot deals with the developmental milestone described in Table 1—"bare" or "blind" double-touch. In this experiment, we used the controller developed in Roncone et al. [18]. The robot is stimulated by touching it on the forearm; see Fig 1 (middle) for a schematic illustration. A modified inverse kinematics solver and controller finds a solution whereby the contralateral fingertip touches the stimulated taxel, and commands both arms to the respective pose (note that the taxel eventually touched by the robot may differ from the one that was initially stimulated because of the systematic errors). Importantly, the robot configuration may differ at each trial, depending on the inverse kinematics solution found by the solver. After the double-touch event, a buffer is used for data collection and learning as explained in Section Data collection for learning. That is, the kinematic model and the joints configuration at every time step are used to convert the position of the tip of the index finger (the approaching body part) to the FoRs of the taxels on the approached and eventually touched part. Unfortunately, only a subset of the skin is physically reachable by the robot—some configurations are kinematically not feasible or unsafe. Therefore, for our experiments, we selected eight taxels (as explained in Section Artificial skin) on the inner part of the forearm for which the double-touch behavior was triggered. These eight taxels updated their representations in parallel using the distance and expected time to contact as the contralateral finger was approaching. As detailed in Table 5, there were 82 successful double-touch trials, with a total of 3512 training samples. That is, there were 82 trajectories sampled at $T = 50ms$ that resulted in a contact with the selected area of the skin. From the eight taxels considered, only six were actually touched at least once by the contralateral index finger. In all of them, the results after learning were qualitatively similar and matched the predictions of our model. The results for one of the taxels with the largest number of training samples (taxel nr. 2; 1625 samples) are shown in Fig 4 and, in fact, they demonstrate learning of a tactile-motor margin of safety: i.e. prediction of self-collisions in the absence of visual input. No offset in the position is reflected in the learned representation, indicating that the model of the kinematic loop connecting the two arms was reasonably accurate.

## Tactile-visual learning

With respect to visual learning, two experiments were performed: (i) the double-touch scenario was repeated, but in this case, utilizing visual input rather than the "motor" information of the

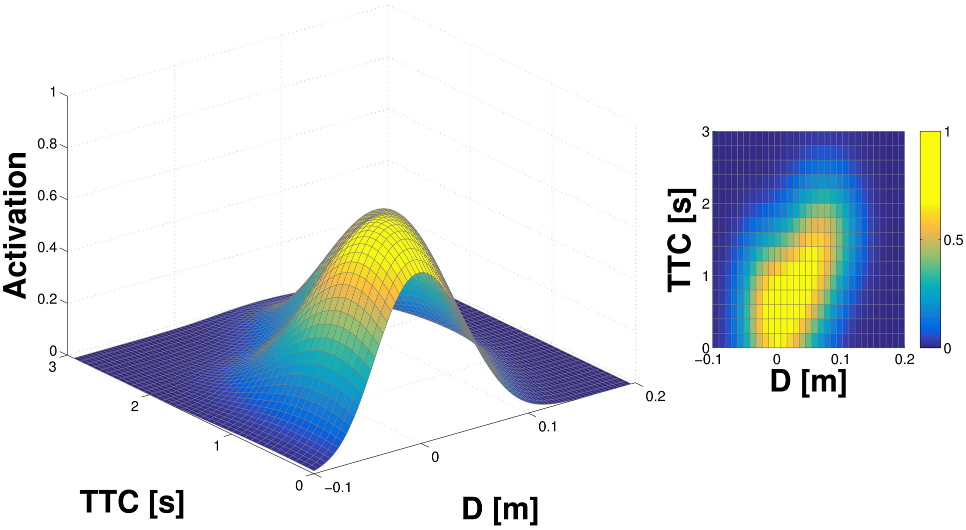


**Fig 4. Tactile-motor representation learned in the double-touch scenario.** Results for taxel nr. 2 on the inner part of the left forearm. See text for details.

doi:10.1371/journal.pone.0163713.g004

moving arm (corresponding to the milestone in Table 2); and (ii) independently moving objects nearing the robot's body were used (Table 3). In both cases, the stimulus (robot fingertip or the moving object) was detected, tracked and its trajectory prior to contact recorded. The position and velocity of the stimulus was extracted and remapped first into the iCub Root FoR and eventually into the FoR of individual taxels, yielding the [*D*, *TTC*] pairs used for learning the representation of nearby space in the corresponding taxels.

**Tactile-visual learning from double-touch.** For this variant of the scenario—double-touch with the moving finger perceived visually—we added a small colored marker to the fingertip that was commanded to execute the double-touch movement. The method to extract the finger's coordinates is described in Section Visual processing and gaze control—"Tracking of fingertip with colored marker". The learning procedure was exactly the same as in the double-touch scenario described earlier. We performed 45 trials. The results show a similar pattern to the previous case; the same taxel (nr. 2; 376 samples) on the inner forearm is selected for illustration in Fig 5.

**Tactile-visual learning using external objects.** This case is a generalization of the double-touch experiments whereby the stimuli are generated by visually perceiving an approaching object that eventually touches the body surface. In this session, tactile-visual trials are carried out by a human experimenter that manually approaches the robot's skin with a series of objects. The visual processing pipeline is explained in Section Visual processing and gaze control—"Tracking of generic objects". This setup was validated using two objects, a cube and a small ball (see Fig 6), approaching the taxels on the robot's body. Importantly, we were no longer limited to parts of the skin that can be activated in the self-touch configurations. We have extended learning to the outer part of the left forearm as well as palm of the right hand.

On the inner part of the left forearm, the same eight taxels of the previous scenarios were considered. Additionally, four taxels on the outer part of the forearm and four taxels of the

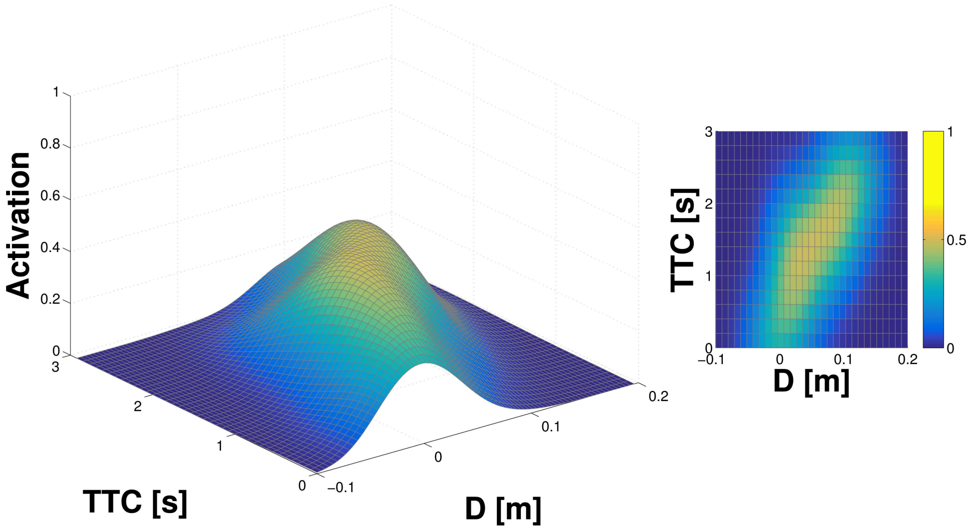


**Fig 5. Tactile-visual representation learned in double-touch scenario.** Results for taxel nr. 2 on the inner part of the left forearm. See text for details.

doi:10.1371/journal.pone.0163713.g005

right palm were also stimulated. We conducted a total of 53 trials for the inner part of the left forearm (events from both objects together), 34 trials for the outer part of the forearm, and 77 trials for the right hand. The results are shown in Fig 7 with the inner part of the left forearm on the left (627 samples, taxel nr. 2), the outer part in the center (451 samples; taxel nr. 8), and right hand on the right (944 samples; taxel nr. 2).

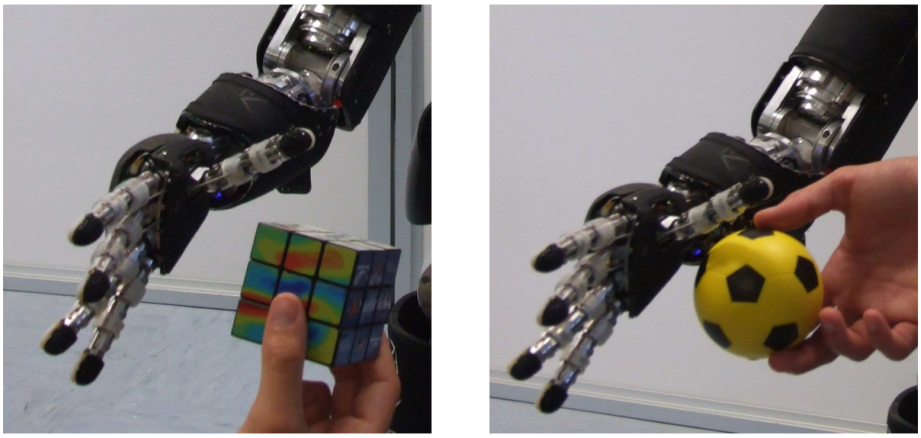


**Fig 6. Objects approaching right palm. (Left)** Cube. **(Right)** Small ball.

doi:10.1371/journal.pone.0163713.g006

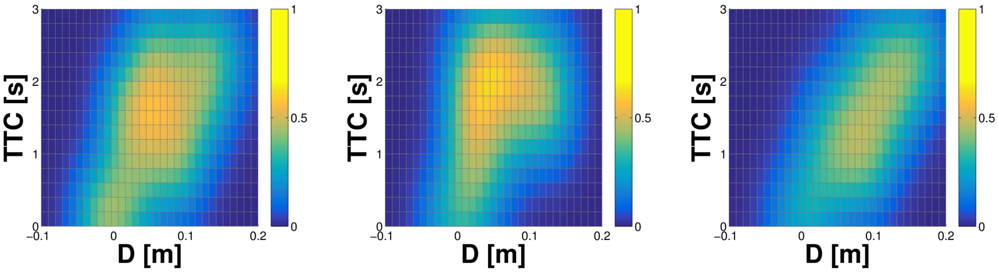


**Fig 7. Tactile-visual representation learned from oncoming objects. (Left)** Inner part of left forearm (taxel nr. 2). **(Middle)** Outer part of left forearm (taxel nr. 8). **(Right)** Right hand (taxel nr. 2). See text for details.

doi:10.1371/journal.pone.0163713.g007

## Interim discussion on learning in the real robot

**Comparison of representations learned in different scenarios.** The experimental results detailed in the previous sections show comparatively similar outcomes for the representations learned on the same taxel (taxel nr. 2 of the internal part of the left forearm) subject to the different experimental conditions (tactile-motor, tactile-visual with double-touch and tactile-visual with external objects). However, there are some differences that are worth mentioning. Specifically, the representation learned in the tactile-motor scenario (Fig 4) shows a "crisper" landscape, which becomes progressively less defined in the subsequent sessions (Figs 5 and 7). This result is expected: as we demonstrated in Section Learning in a single taxel model, an increase of the noise in the input signal as well as in the variability of the stimulation results in a broader profile of the activation landscape (see Figs 2 vs. 3). The double-touch (tactile-motor) scenario is a highly controlled setup in which the robot performs a number of similar trials with similar velocity profiles, using an inverse kinematics solver and controller. By reducing reliance on the kinematics, and progressively depending on an intrinsically noisy sensory system (i.e. the visual system), the contribution of noise becomes more prominent. Further, training trials for the tactile-visual learning with external objects are performed by a human experimenter, with little control on the type of trajectories that are presented to the robot, resulting in a broader landscape of the probability function.

**Comparison of representations learned by different body parts.** In the Tactile-visual learning using external objects scenario, three different skin parts were subject to training: internal and external part of the left forearm, and the right hand (palm). Representations learned around selected taxels were shown in Fig 7. Here we look at aggregate statistics of all the taxels for each of the skin parts. We postulate that a significant component of the systematic error pertaining to a taxel is skin part specific and can be mainly attributed to the position on the kinematic chain (e.g. forearm vs. hand) and the mounting of individual skin patches (see Experimental Setup). In order to validate this hypothesis, we extrapolate the systematic offset of the ten virtual taxels that were stimulated during the experimental session and analyze the overall trend between different body parts. To this end, we performed a weighted orthogonal 2D least-squares regression, with $[x, y]$ coordinates given by $[D, TTC]$, and weights equal to the learned representation at each of the pairs (i.e. the contact probability, $f(D_i, TTC_j)$, see Eq 3 under Internal representation). A weighted 2D regression applied to the 3D landscape reduces the dimensionality of the input space, and lets us evaluate at which distance $D$ the regression line crosses the $x$–axis (i.e. $TTC = 0$)—giving the offset pertaining to the position of the particular taxel. Results are depicted in Fig 8: most of the taxels show an overall error between 1cm

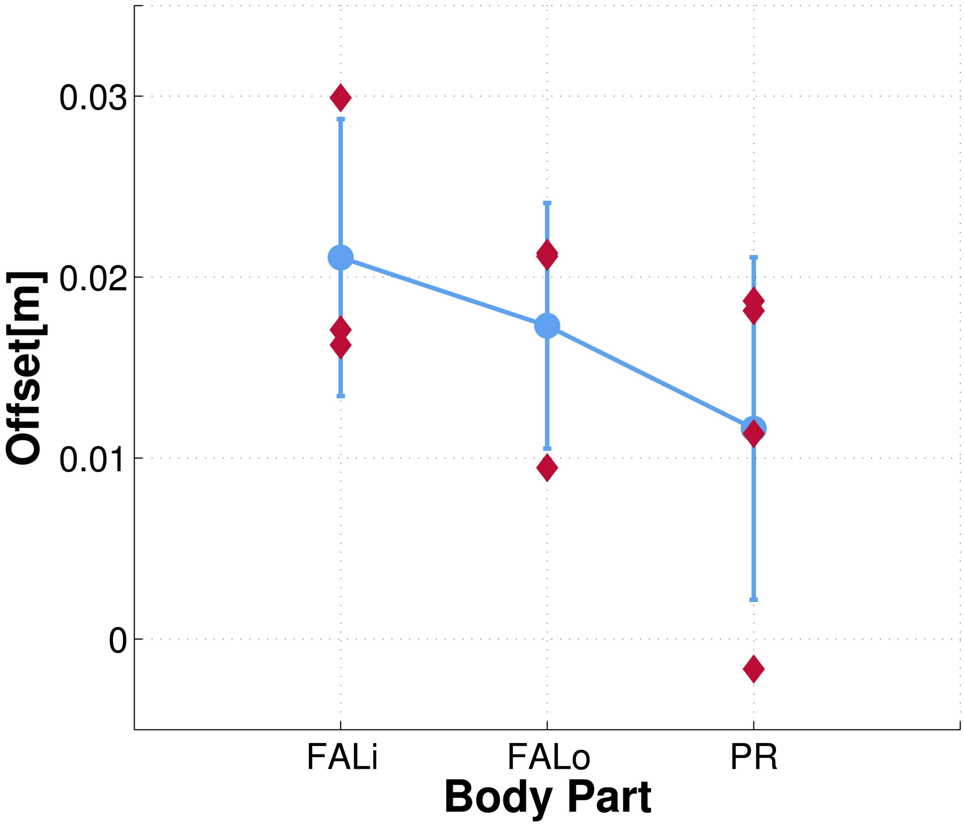


**Fig 8. Systematic offsets computed during tactile-visual learning using external objects.** The distance offset in the learned representation of ten taxels (three on the inner part of the left forearm, *FAL*$_i$; three on the outer part, *FAL*$_o$; four in the right palm, *PR*) is depicted in red. For each of the three body parts under consideration, average offset and standard deviation are depicted in blue.

doi:10.1371/journal.pone.0163713.g008

and 3cm, with an average error of 2.11cm and 1.73cm for the inner and outer part of the left forearm respectively, and 1.16cm for the right palm. The results suggest that the systematic errors depend on the specific skin part the taxels belongs to, even though additional "intra-skin-part" variance is present. Importantly, the learned representations automatically compensate for these errors as will be demonstrated later.

**Analysis of the learning progress.** As mentioned in Section Learning in the real robot, one of the features of the model is the ability of each taxel to learn a usable representation very quickly, from a few training samples. This is a direct consequence of the smoothing approach (Parzen windows applied to the discrete representation) for undersampled spaces. To illustrate this, in Fig 9 we show the evolution of the representations belonging to two neighboring taxels in the internal part of the left forearm during tactile-visual learning with external objects. Starting from a blank state for both taxels, we depict the representation after the same "positive"

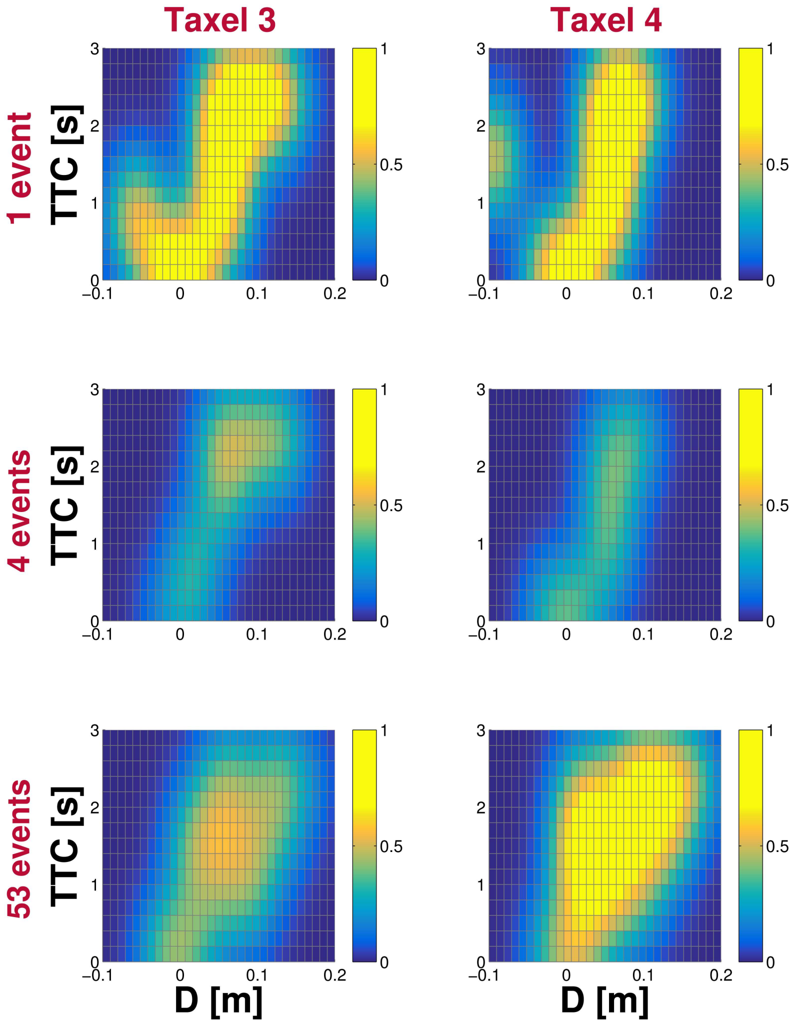


**Fig 9. Evolution of the learning process.** The progress of the learned representations belonging to two adjacent taxels of the internal part of the left forearm is shown. For each of the two taxels (taxel nr. 3 on the left column and taxel nr. 4 on the right column), snapshots of their respective representations after 1, 4 and 53 trials are depicted. See text for details.

doi:10.1371/journal.pone.0163713.g009

example (i.e. the nearing object contacted both taxels), after 4 examples (combination of positive and negative trials), and after the full training of 53 examples.

The results show how the same input trial (approaching stimulus) affects each taxel differently, because it gets projected on each taxel's FoR in a slightly different manner. After the first trial, there is a clear bias toward the only experience the taxels had (the lack of negative examples practically translates into maximal certainty of collision prediction for some parts of the input space). Nonetheless, although the representations are far from being comparable to their respective final versions, even with a single (positive) example they can be already used for a coarse estimation of the probability of being touched by future incoming objects. Finally, after the full training session, the respective landscapes of the two adjacent taxels show similarities in both their shapes and offsets. Yet, taxel nr. 4 exhibits stronger responses over its landscape, which is a consequence of the fact that taxel nr. 3 is positioned closer to the robot's wrist and it is thus less likely to be contacted and to experience positive trials. This illustrates the effect of the individual taxel's training on the learned representation, which is further shaped by the embodiment—the taxel's physical placement in this case.

## Exploitation of the learned associations

The learned representation is validated during an avoidance/reaching experimental session, corresponding to the last milestone: exploitation of learned associations (Table 4). The robot uses the acquired model in order to either avoid or come into contact with an incoming stimulus with any of the skin parts that have a peripersonal RF. Similarly to the learning stage, experiments are conducted by presenting the robot with a series of stimuli. An approaching object thus triggers the activation of each taxel given by the taxel's previous "experience" with similar stimuli (in terms of $[D, TTC]$). Consequently, this gives rise to a distribution of activations pertaining to the skin surface. It is important to note the following: i) the iCub built up a PPS representation based on stimuli that are directed toward the skin; ii) in order to test these representations, we exploit a similar scenario, in which the robot has to either move away from or reach for approaching objects. Static objects (or objects that are moving *away* from the robot) do not trigger a response from the PPS representation and hence do not generate any movement, which is desirable and in accordance with neurophysiological data on approaching vs. receding stimuli (see e.g., Graziano and Cooke [2]).

The iCub is presented with an unknown object that was not used in the learning stage (a pink octopus). It is used by the experimenter to perform a series of approaching behaviors toward the robot's body parts that had previously learned their representations (left forearm and right hand). The visual processing pipeline used was identical to the learning stage (see Section Visual processing and gaze control). However, here, the taxels' activations are exploited by the robot to either avoid or "reach for" the approaching object with any of the body parts used during learning. Only taxels with activation above a certain threshold contributed to the resulting movement vector that was eventually executed by the controller. The threshold was empirically set to 0.4, corresponding to a 40% chance of that taxel being contacted by the nearing object (according to the learned model). In order to achieve the desired behavior, we implemented a velocity controller that can move any point of either the left or right kinematic chain of the arms in a desired direction. During an avoidance task, the movement is directed away from the point of maximum activation, along the normal to the local surface in that point. For "reaching", the desired movement vector has the opposite direction. The setup of the controllers is described in Section Avoidance and reaching controller.

**Margin of safety: Avoidance behavior demonstration.** To demonstrate the performance of the avoidance behavior, we conducted an experimental session of roughly 20 min. in duration where we performed a series of approaching movements with a previously unseen object, the octopus toy, alternating between the body parts and varying the approaching direction. Avoidance behavior was successfully triggered in all cases. A snapshot illustrating typical behavior in a 15*s* window for the left forearm (Fig 10 left) and a 20*s* window for the right palm (Fig 10 right) is shown—with two approaching events in each plot. In total, nine taxels of the

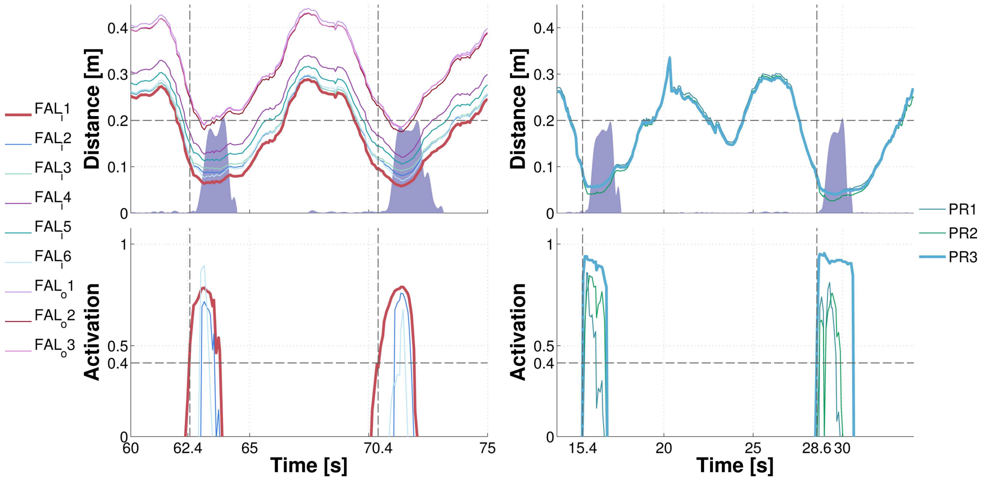


**Fig 10. Avoidance demonstration. (Left)** Object approaching the inner part of left forearm. Nine taxels of the left forearm (six on the inner part, $FAL_i$ 1: 6; $FAL_i$ stands for forearm left internal; three on the outer part, $FAL_o$ 1: 3) were considered in the experiment. Top plot shows the distance of the object from the taxels in their individual FoRs. The shaded purple area marks the velocity of the body part (common to all taxels; maximum activation corresponding to 10$cm/s$). Bottom plot depicts the activations of the forearm taxels' PPS representations. **(Right)** Object approaching the right palm. There were three taxels considered ($PR$ 1: 3, where PR stands for "palm right").

doi:10.1371/journal.pone.0163713.g010

left forearm (six on the inner part; three on the outer part) and three taxels of the right palm were considered. The top plots show the distance of the approaching object from the individual taxels (in their respective FoRs). The bottom plots show the activations of the learned representations for each taxel (note that this representation uses a two-dimensional domain of [$D$, $TTC$]; however, to demonstrate the behavioral performance, we restrict ourselves to showing distance only in the upper plot). As the object comes closer, there is an onset of activation in the "most threatened" taxels. Once the activation level exceeds a predefined threshold (0.4 in this case—horizontal line in the bottom panels), the avoidance behavior is triggered. This is illustrated in the top plots with the shaded purple area that marks the velocity of the body part as a result of the avoidance controller command. The first taxel responding is highlighted in the corresponding upper and lower plots. The upper plots clearly demonstrate that the avoidance behavior was effective: a safety margin has always been preserved. Qualitatively similar behavior was observed during the whole experimental session. The controller takes advantage of the distributed representation pertaining to individual taxels, averaging the expected contact locus as well as its likelihood. This loosely resembles the way noisy information is averaged in neural population coding schemes (e.g., [50]).

**"Reaching" with arbitrary body parts.** In a similar fashion, we tested the "reaching" controller in a session of approximately 10 min. in duration. Note that this is "reaching" not in a traditional sense of reaching with the end-effector—the hand. Instead, the particular skin area most likely to collide with the stimulus will be recruited to "reach" or "catch" it. A snapshot illustrating the performance while approaching the inner part of left forearm is shown in Fig 11. The graphical illustration is the same as in the avoidance case. The spatial representations pertaining to the taxels get activated (bottom plot) and trigger the movement, which in this case is approaching the object. In addition, the bottom plot illustrates the physical skin

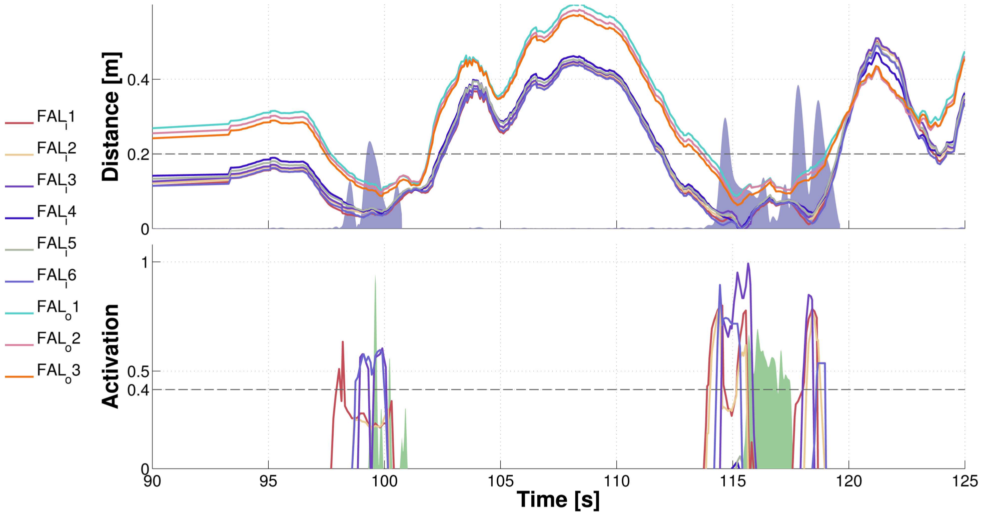


**Fig 11. Reaching with arbitrary body parts demonstration.** Object approaching the inner part of left forearm. Nine taxels of the left forearm (six on the inner part; three on the outer part) were considered in this experiment. Top plot shows the distance of the object from the taxels in their FoRs. The shaded purple area marks the velocity of the body part due to the activation of the controller. The bottom plot depicts the activations of the forearm taxels' representations. The green shaded area marks physical contact with the robot's skin—aggregated activation of all tactile sensors contacted on the body part.

doi:10.1371/journal.pone.0163713.g011

activation (green shaded area). Importantly, contact is generated in both cases as the skin activation testifies. The fact that the distance is greater than zero in the first event can be attributed either to errors in the visual perception or to an offset in the kinematic transformations.

**Joint space and operational space range.** During tactile-motor and tactile-visual training using "double-touch", the robot controls both arms to satisfy the self-touch constraint, thus automatically attaining different—even if somewhat stereotypical—arm configurations. Conversely, the tactile-visual learning using external objects was performed in a static configuration—the robot is passively waiting for external objects to contact the skin. Nonetheless, exactly the same software is used in all cases, since it automatically handles any configuration. The robustness of our approach to different arm configurations is even more evident in the subsequent exploitation of the learned associations, which involved the richest repertoire of configurations. The movement response is always different, depending on where the object is coming from and which portion of the peripersonal space representation is activated the most. To illustrate the range of different configurations, we have extracted in Table 6 the extremes reached by individual degrees of freedom (DoFs). It is evident how most of the joints actively involved in the movements (the shoulder joints and the elbow) have spanned a large portion of their range, with some of them even covering all of their operational range. In addition, we quantified the range of the end-effectors in the operational space (see Table 7, with further details provided in S1 Fig). For safety reasons, the range of the end-effectors during the experiment was artificially restricted to be confined to a sphere of radius 0.2 $m$ around the home position. Also, note that the data was recorded only while the peripersonal space representation was active—i.e. while activations were exceeding a threshold. In summary, this demonstrates

**Table 6. Range of arm DoFs during avoidance and reaching.** Each of the 7 DoFs that belong to the left and right arms are depicted: 3 DoFs for the shoulder ($s_0$, $s_1$, $s_2$), one elbow joint ($e_0$) and three joints pertaining to the wrist ($w_0$, $w_1$, $w_2$). For each joint, its minimum and maximum angles are shown, along with its range. Joints $s_2$ and $e_0$ of the left arm, as well as joint $s_2$ of the right arm, reached their full physical limits during the experiments. Wrist joints did not contribute to either the avoidance or the reaching behaviors.

| Left Arm [deg] | | | | | | | |
|---|---|---|---|---|---|---|---|
| | $s_0$ | $s_1$ | $s_2$ | $e_0$ | $w_0$ | $w_1$ | $w_2$ |
| Min | -65.0 | 19.5 | -39.2 | 16.3 | -0.88 | -0.015 | -0.025 |
| Max | 7.1 | 64.2 | 80.1 | 106.0 | 1.38 | 0.064 | 0.063 |
| Range | 72.1 | 44.7 | 119.9 | 89.7 | 2.26 | 0.079 | 0.088 |
| **Right Arm [deg]** | | | | | | | |
| | $s_0$ | $s_1$ | $s_2$ | $e_0$ | $w_0$ | $w_1$ | $w_2$ |
| Min | -61.7 | 18.6 | -37.8 | 15.5 | -1.57 | -0.079 | -0.085 |
| Max | 7.9 | 45.5 | 80.7 | 80.5 | 0.86 | 0.111 | 0.159 |
| Range | 69.5 | 26.8 | 118.5 | 65.0 | 2.43 | 0.190 | 0.244 |

doi:10.1371/journal.pone.0163713.t006

that the representations learned were robustly activated in a wide range of joint configurations and end-effector positions.

**Comparison with model without TTC information.** In this section, we compare the proposed approach with our previous work [45]. In particular, in this work we benefit from a richer representation because of the introduction of the time-to-contact (TTC) dimension. Although this results in a more complex model and the need to increase the number of training samples in order to converge to a stable representation, we believe that the information carried out by the TTC is crucial in the construction of a model of nearby space that is meaningful and effective in a real world scenario. Specifically, by including dynamic information about the speed of the approaching object, the proposed model can easily distinguish which objects pose an immediate threat to the body. To make a practical example, the TTC of a close but static object would be infinite, whereas it would be negative for an object that is moving away from the skin; in both cases, such objects would be easily discarded because they would not fall within the boundaries of our representation, which considers objects with a TTC included in the range $[0; 3s]$. The exploitation of this feature can be demonstrated by comparing the avoidance and "reaching" controllers in this work and [45]. Without loss of generality, in the following we compare only the avoidance behaviors, although similar conclusions can be drawn by analyzing the "reaching" with arbitrary body parts controllers. Fig 10 shows how the taxel of interest is activated only when the object is approaching it, i.e. when its distance decreases over time. When the object is moved away by the experimenter (approximately at $t = 64s$ and $t = 71s$ in Fig 10 left), the taxels become silent and the avoidance behavior stops. A comparison with previous work—see S2 Fig—, instead, shows how this is not the case if only the distance is

**Table 7. End-effector extremes in operational space during avoidance and reaching.** For both the left and right end-effectors, the minimum and maximum values reached in the $x-$, $y-$ and $z-$ axis are shown, along with its range of operation. For safety reasons, the operational space of the robot was constrained within a sphere centered in the resting position (set to [−0.30, −0.20, +0.05] *m* for the left arm and [−0.30, +0.20, +0.05] *m* for the right arm—in iCub Root FoR) and with radius equal to 0.2 *m*. Please refer to S1 Fig for a depiction of the robot's kinematics during the avoidance and reaching scenario.

| Left End-Effector [m] | | | | Right End-Effector [m] | | | |
|---|---|---|---|---|---|---|---|
| | *x* | *y* | *z* | | *x* | *y* | *z* |
| Min | -0.34 | -0.39 | -0.06 | Min | -0.34 | 0.06 | -0.03 |
| Max | -0.18 | 0.00 | 0.15 | Max | -0.19 | 0.36 | 0.10 |
| Range | 0.15 | 0.40 | 0.22 | Range | 0.15 | 0.29 | 0.13 |

doi:10.1371/journal.pone.0163713.t007

taken into account: the taxels' activation fades completely only if the object moves away enough to fall out of the receptive field, i.e. farther than 20*cm* from the skin.

## Discussion and Conclusion

In this paper, to the best of our knowledge, we presented the first robot that learns a distributed representation of the space around its body by exploiting a whole-body artificial skin and either self or environment physical contact. More specifically, each tactile element has been associated to a spatial receptive field extending in the 3D space around the skin surface. Stimuli in the form of motor or visual events are detected and recorded. If they eventually result in physical contact with the skin, the taxels update their representation tracing back in time the approaching stimulus and increasing the quality of the internal probability estimate—in terms of distance and time to contact—that is, an estimation of the likelihood that the stimulus eventually touches any given body part. The spatial RF around each taxel is constructed and updated as the limbs move in space by combining the joint angles and knowledge of the robot's kinematics; however, its representation is adapted from experience, thus automatically compensating for errors in the model as well as incorporating the statistical properties of the approaching stimuli. This representation naturally serves the purpose of predicting contacts with any part of the body of the robot, which is of clear behavioral relevance. Furthermore, we implemented an avoidance controller whose activation is triggered by this representation, thus endowing the iCub with a "margin of safety". Finally, simply reversing the sign of the controller results in a "reaching" behavior toward approaching objects, using the closest body part.

One important feature of the proposed method is its invariance with respect to the robot configuration (posture) and the input modality used. Capitalizing on the robot's kinematic model, current stimulus positions are automatically remapped into every taxel's FoR, taking also every taxel's current pose (position and orientation) into account. In the double-touch scenario, both the "receiving" arm with the taxel array and the "touching" arm with the extended finger (the nearing stimulus) move, which, however, does not pose any difficulty for our method. Furthermore, our model is agnostic as to whether the stimulus was perceived motorically or visually. In the last scenario with external approaching objects, the arm configuration was static during learning, but the head and eyes were moving. Nevertheless, a moving arm would again be automatically considered using exactly the same computation. This is also demonstrated in the avoidance / "reaching" scenarios, where the arm moves, but the stimulus' effect on the taxels is constantly evaluated, resulting in online adaptation of the robot response.

Another important asset of the proposed model is that learning is fast, proceeds in parallel for the whole body, and is incremental. That is, minutes of experience with objects moving toward a body part produce a reasonable representation in the corresponding taxels that is manifested in the predictive prior to contact activations as well as in the avoidance behavior. Smoothing using Parzen windows applied to the discrete representation specifically contributes to this effect in the case of undersampled input spaces.

The investigated scenarios parallel those experienced by humans and animals—also because of the anthropomimetic nature of the iCub—and should thus inform us directly about the mechanisms of peripersonal space representations in primates as they have been subject of intensive investigations in cognitive psychology as well as the neurosciences over decades. First, the developmental trajectory leading to the acquisition of these representations is largely unknown. The development of reaching (e.g., [51, 52]) may constitute one key factor in this mechanism; the exploration of own body may be another (e.g., [11, 17]). In this paper, we mimicked a similar developmental trajectory by considering first the self-touch behaviors and adding encounters with objects later on.

The architecture presented is, at this stage, not a model of a particular brain network. Casting it into the vocabulary common in the neurosciences, one could say that the representation associated with every taxel may correspond to a spatial receptive field of a neuron that is centered on that particular taxel (hence body-part centered coordinates). The RF has a fundamentally spatial nature; further, it is modality-independent—as we demonstrated by entering it and eliciting its "neural" response with motor/proprioceptive as well as visual targets. However, note that this "neuron" does not have a tactile RF—tactile sensations were used in the learning/adaptation of this RF only. However, it would be easy to extend our representation by constructing a bimodal visuo-tactile or, more precisely, tactile-spatial neuron whose activation would be the sum of the "spatial" and tactile inputs. The reference frame transformations are in our case mediated by the kinematic model of the robot and use the iCub Root FoR as common ground connecting all kinematic and visual chains. This is unlikely to correspond to the exact mechanism used by the brain; however, bimodal neurons with tactile RFs on a body part and visually RFs around it and anchored to it—following it in space independently of eye position—have been identified both in premotor areas (F4) (e.g., [53] for a survey) and parietal areas (VIP and other—e.g., [23]). Our position is similar to [29], for example, assuming that the necessary coordinate transformations (from visual or proprioceptive input to body-part centered coordinates) are performed by an upstream process. Our model then receives this as input. Several common reference frames (e.g., eye-centered [54]) have been proposed to act in the posterior parietal cortex. In summary, the architecture presented is a first implementation that supports the relevant behaviors. However, since the scenarios as well as the sensory modalities available to the robot parallel the conditions in biology (at a certain level of abstraction), the road is open to further grounding of the architecture to the corresponding putative brain mechanisms.

One possible practical limitation of the presented architecture is its computational and memory requirements. The distributed and parallel nature of the representation has many advantages. At the same time, the complexity grows linearly with the number of taxels—each of them monitoring its spatial receptive field and, possibly, updating its probabilistic representation. However, this is clearly in line with the nature of brain computation. Furthermore, the spatial resolution we have selected (with taxels of around $2cm$ in diameter on the skin surface) is likely unnecessarily high—the body-part-centered receptive fields of parietal cortex neurons are typically much larger (e.g. spanning a whole upper arm [5]). Also, lower resolution may still suffice to support the margin of safety behavior. Such a modification would be straightforward in our setup, requiring only a redefinition of the virtual taxels.

The "demonstrators"—avoid ance and "reaching"—are also only first steps in this direction. They are simply exploiting fairly standard controllers to generate movements of a virtual point that is a result of voting of taxels activated by a moving object. Avoidance differs from reaching in the direction of this movement vector only. This could be further differentiated and developed, leading to simple reflexive as well as complex whole-body avoidance mechanisms such as those reported in monkeys [2]; an implementation in the iCub relying on force/torque feedback has been presented in [41]. Finally, "reaching" here is a simple mechanism that results in approaching to a nearing object with the skin part that was most likely to be contacted by the object. Yet, this resembles the principle of motor equivalence, where the controller in fact can generate reaching movements using arbitrary body parts as end-point.

Future work can proceed along several directions. First, the architecture can be refined and better grounded in concrete mechanisms that are assumed to operate in the primate brain, leading to a better explanation of why certain connectivity patterns including polymodal neurons are a necessity and not only the result of the quirks of evolution. This would provide an invaluable tool to test biological theories and crucially advance the computational modeling efforts. Second, the full kinematic model of the robot that was taken for granted here could be dropped and the learning problem expanded to full complexity dealing with the emergence of spatial representations from motor, proprioceptive, tactile and visual inputs. The double-touch scenario could in fact serve this very purpose of body schema learning; the self-calibration framework of [18] could be adapted and the Denavit-Hartenberg representation of kinematics and the inverse kinematics solver replaced by more biologically motivated analogs. Third, the margin of safety in primates does not have uniform extension and resolution; instead, body parts, in particular face and hands, receive more attention than others. This could be emulated in the robot as well. Fourth, the model proposed in this work could be further developed to address the expansion of the RFs after tool use as first documented by [34] and modeled by [33] in a humanoid robot. Fifth, the architecture proposed is prone to impact on practical applications. Whole-body tactile sensing together with a virtual margin of safety around the robot's body dramatically increases the robot's own safety as well as safety of humans that share the environment with the robot. The proposed implementation will have to be tested in such scenarios and possibly enhanced also by force/torque sensing that is already available on the iCub to guarantee robustness in all situations. Finally, with the advent of robotic skin technologies (see [39] for a review), frameworks similar to this one can find applications in diverse robotic platforms and are by no means restricted to the iCub humanoid robot (or to humanoids altogether).

## Materials and Methods

### Experimental Setup

The iCub is a full humanoid robot platform originally developed to support research in artificial cognitive systems. In this section we describe the key components relevant for this work: the artificial skin, the robot's sensing modalities, the eye and camera setup, the model of the robot's kinematics, visual processing and gaze control, and finally the avoidance and reaching controller used in the experiments. For details on the basic structure of the iCub we refer the reader to [55].

**Artificial skin.** Recently the iCub has been equipped with an artificial pressure-sensitive skin covering most body parts [26]. The latest iCub version contains approximately 4000 tactile elements (taxels)—in the fingers, palms, forearms and upper arms, torso, legs and feet. In the experiments performed in this work, we restrict ourselves to the forearms and palms. The iCub forearm and hand with exposed skin is shown in Fig 12 left. With the exception of the palm and fingertips, the skin covering all body parts consists of patches with triangular modules of 10 taxels each (Fig 12 middle). There are in total 23 modules on the forearm in two patches and hence 230 taxels. However, for the purposes of this study this resolution would be an unnecessary burden. Therefore, we generate RFs grouping all responses in a triangular module in a single "virtual" taxel whose position in the body surface corresponds to the central physical taxel.

The palm has a slightly different structure corresponding approximately to four triangles (see Fig 12 right). It is made out of a single printed circuit board composed of an array of 43 taxels. We artificially split the array into four regions of 8 to 10 taxels, forming virtual taxels as before. These are shown in Fig 12 (right), with the central taxels marked with full circles. The region enclosed between the thumb and the fingers is not considered as it is unlikely that it is touched by a moving object. In the main article, we use "taxel" to refer to this virtual taxel.

A spatial calibration of the skin of the forearm with respect to the iCub kinematic model has been performed in del Prete et al. [48]. The palm was calibrated using data from the CAD

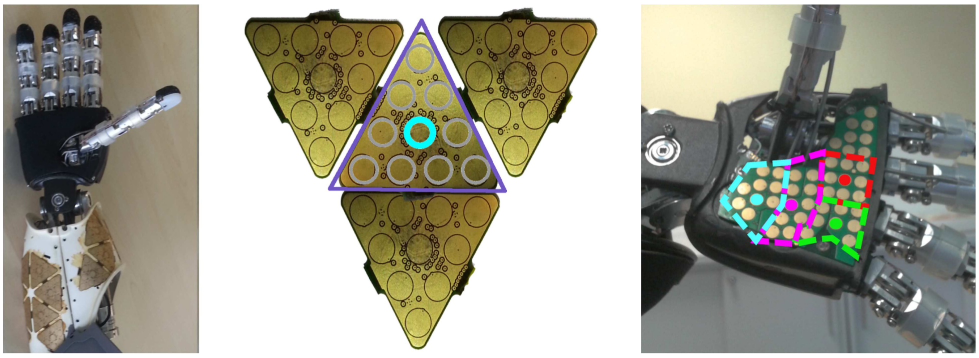


**Fig 12. Pressure-sensitive skin of the iCub. (left)** iCub forearm with exposed skin patches. **(middle)** Four triangular modules of the iCub skin PCB (10 taxels each). The central taxel corresponds to the virtual taxel which in turn is made by joining the responses of a full triangle. **(right)** Exposed skin of the palm with virtual taxels highlighted.

doi:10.1371/journal.pone.0163713.g012

model. The position of all taxels as well as the orientation of their surface normal in the iCub Root FoR can thus be extracted if the current joint configuration is known.

**Joint angle sensing.** Proprioceptive inputs in the iCub simply consist in angular position measurements at every joint. For most joints, they are provided by absolute 12bit angular encoders (see [55] for details); small motors (head and hands) employ incremental encoders whose zeros are calibrated at startup.

**Head and eyes.** Vision of the iCub is provided by two cameras mounted in the robot's eyes. The neck of the robot has three degrees of freedom (DoFs) and there are three additional DoFs in the eyes allowing vergence and version behaviors. The tilt DoF is mechanically coupled (both eyes move up and down); the version and vergence movements are coupled in software following an anthropomimetic arrangement. With appropriate calibration [56], depth information can be extracted from binocular disparity.

**Kinematic model and coordinate transformations.** The iCub sensors provide raw data in different FoRs. These need to be transformed in order to compare similar quantities. In primates, the role of establishing a common ground between these rich but diverse sources of information is attributed to the body and peripersonal space representations. As we described in Section Introduction, coordinate transformations (such as between eye-centered and body-part-centered FoRs) are necessary. In our case, we specifically need two types of transformations:

- *Purely kinematic transformations.* For the first scenario where the robot learns about the space around its body through double-touch (cf. Table 1), the "touching" body part (like the index finger of the right hand—Fig 1 middle) need to be brought to the FoR of the taxels that are "touched" (like the skin on the left forearm).
- *Visual-kinematic transformations.* In the second scenario, vision is considered. There are two variants of the experiment: i) double-touch with visual tracking of the finger approaching the contralateral arm (see Table 2); ii) an independently moving object approaching and touching the robot's skin (cf. Table 3). In both cases, transformations from the image (retina) FoR are necessary. This involves exploiting binocular disparity to obtain a 3D position of the object (or finger) in the head FoR and then following a sequence of coordinate transformations to eventually reach the FoR of individual taxels.

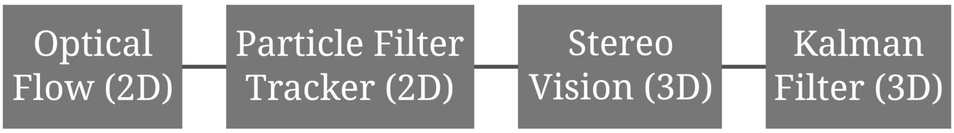


**Fig 13. Tracking of generic objects.** See text for details.

doi:10.1371/journal.pone.0163713.g013

Learning these transformations was not the goal of this work; therefore, we have employed the existing kinematic model of the iCub that is based on the Denavit-Hartenberg convention and available in the form of a software library (*iKin*, [57]). The library allows traversing any kinematic chain of the iCub by employing an appropriate sequence of transformations. In fact, kinematic representations of individual chains in iKin start/end in the Root FoR of the robot (around the robot's waist) and this is employed as an intermediary to connect individual sub-chains. The transformation to individual taxels' FoRs is provided by the skin calibration.

These composite transformations are subject to errors that include (i) mismatch between the robot model based on the mechanical design specifications (CAD model) and the actual physical robot; (ii) inaccuracies in joint sensor calibration and measurements; (iii) unobserved variables as for example backlash or mechanical elasticity; (iv) inaccuracies in taxel pose calibration; (v) errors in visual perception due to inaccurate camera calibration. The combination of these sources of error can amount to a total of several centimeters. However, in the proposed approach, the representation that every taxel learns with regard to its surrounding environment will automatically compensate for the systematic component of these modeling errors.

**Visual processing and gaze control.** For the scenarios involving tactile-visual learning (described in Tables 2 and 3), additional processing steps are needed to compute the approaching stimuli's position and velocity: moving objects need to be detected, segmented out of the background and their position tracked. We implemented two variants of the tracking mechanism:

1. **Tracking of fingertip with colored marker**. In this case, we implemented an HSV-based color segmentation module that can track a green marker placed on the iCub's index fingertip on both the right and left image. A simple triangulation procedure yields the 3D coordinates of the fingertip in the robot's Root FoR.
2. **Tracking of generic objects**. In this second case, we used a tracker for generic objects under certain moderate assumptions on the availability of visual features and limits on their velocity and size, as developed in Roncone et al. [45]. The tracking software consists of a number of interconnected modules, schematically depicted in Fig 13. The first module uses a 2D Optical Flow [58] to detect motion in the image. If this is the case, it triggers a 2D particle filter module [59] to track the object in the image plane based on its color distribution. At this stage, the 2D planar information related to the approaching object (namely, the centroid of the object and an estimation of its size) is converted into 3D (world) coordinates via a stereo disparity module [60]. A Kalman filter then completes the position estimation process. It uses 3D points as determined by the stereo vision module and it employs a fourth order dynamical model of the object motion. Finally, a gaze controller was employed in order for the eyes and head to smoothly follow the tracked object in space. The details of the gaze controller can be found in [20].

## Data collection for learning

As outlined above, two distinct scenarios were considered where a given skin patch was stimulated by the robot's own body (double-touch) or by independently moving objects. However, the basic principle is the same, that is, in both cases it is implemented in a local, distributed, event-driven manner. An illustration of the two cases is depicted in Fig 1 (middle and right). A volume was chosen to demarcate a spatial "receptive field" (RF) around each taxel (we will use this notion of receptive field for the scenario in the robot from now on). Similarly to what happens in humans and monkeys, these receptive fields distributed around the body are anchored to the body part they belong to and encode local information. However, unlike in biology where receptive fields of individual neurons are tied to a particular sensory modality and response properties of the neuron, our receptive field is a theoretical construct—a volume of space around the taxel. In what follows, any stimulus moving toward the robot's body—note that this can be either another part of the body of the robot or an external visual stimulus—will be remapped into the taxel's reference frame and thus potentially enter its receptive field. The RFs are limited to a conical volume oriented along the normal to the local skin surface and extend to a maximum of 20cm from the surface (green region in Fig 14). This is consistent

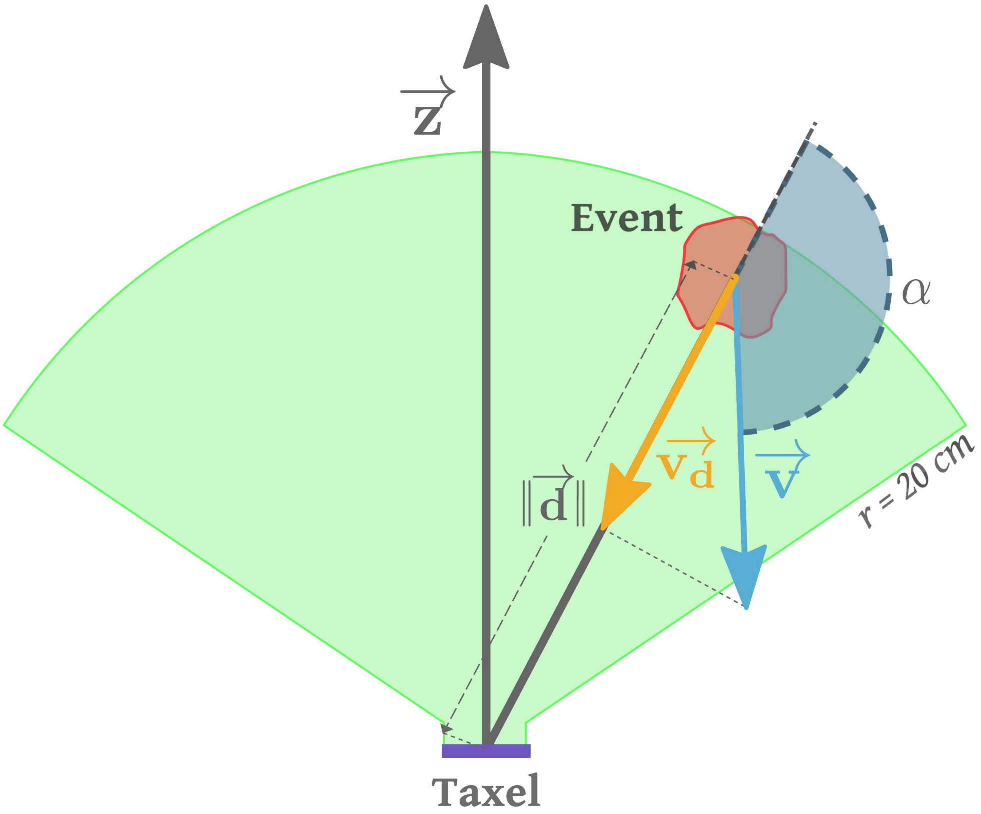


**Fig 14. Receptive field of a taxel and approaching stimulus.** See text for details.

doi:10.1371/journal.pone.0163713.g014

with neurophysiological observations [2]. When a stimulus enters the conical volume of a RF, we mark the onset of a potentially interesting event. Subsequently, the position and velocity of the object w.r.t the taxel is recorded and the distance *D* and time to contact *TTC* computed. The scalar distance, *D*, is calculated as follows:

$$D = \operatorname{sgn}(\vec{\mathbf{d}} \cdot \vec{\mathbf{z}}) ||\vec{\mathbf{d}}|| \quad , \tag{1}$$

where $\vec{\mathbf{d}}$ is the displacement vector pointing from the taxel to the stimulus (geometric center of the incoming object), $\vec{\mathbf{z}}$ is the *z*-axis of the reference frame centered on the taxel and pointing outward (coincident with the normal to the skin surface at the taxel position). The sign of the dot product is positive if the object lies in the hemisphere extending from the taxel. The scalar distance, *D*, preserves information about the relationship of the event w.r.t. taxel normal. *D* can be negative because of modeling or measurement errors or simply because a stimulus is behind a particular body segment. The time to contact, *TTC*, is defined as follows:

$$TTC = -\operatorname{sgn}(\vec{\mathbf{d}} \cdot \vec{\mathbf{v}}) \frac{||\vec{\mathbf{d}}||}{||\vec{\mathbf{v}}_{\mathbf{d}}||} = -\operatorname{sgn}(\vec{\mathbf{d}} \cdot \vec{\mathbf{v}}) \frac{||\vec{\mathbf{d}}||}{||\vec{\mathbf{v}} \cdot \cos(\alpha)||} \quad , \tag{2}$$

where $\vec{\mathbf{d}}$ is again the displacement vector pointing from the taxel to the stimulus, $\vec{\mathbf{v}}_{\mathbf{d}}$ is the projection of the stimulus's velocity $\vec{\mathbf{v}}$ onto $\vec{\mathbf{d}}$, and $\alpha$ is the angle between $\vec{\mathbf{v}}$ and $\vec{\mathbf{d}}$, as shown in Fig 14. The sign term takes into account the direction of motion of the stimulus. That is, for stimuli in the "positive hemisphere" moving toward the taxel, the dot product will be negative ($\vec{\mathbf{d}}$ and $\vec{\mathbf{v}}$ have opposite directions) and the time to contact will be positive. The opposite holds for objects moving away from the taxel or the case when modeling errors return a negative distance. The magnitude of *TTC* is simply distance over speed (norms of the respective vectors, $\vec{\mathbf{d}}$ and $\vec{\mathbf{v}}_{\mathbf{d}}$).

This definition of *D* and *TTC* is clearly an approximation that simplifies the estimation of probability densities by bringing down the full description of a stimulus motion into a bi-dimensional space. This is useful to keep the learning procedure feasible with a small number of data points and it has the additional advantage of allowing one-shot learning: a single stimulus and contact with the skin enables a rough but useful estimation of the corresponding RF density. Note that this procedure—the recording of *D* and *TTC* of approaching stimuli—is carried out in parallel for all taxels whose RFs overlap with the stimulus location. These data are buffered for three seconds and used for learning only if the stimulus eventually touches the skin and at least one taxel measures the contact event. In this case, a learning iteration is triggered as follows:

1. For all the taxels that measured a contact event, the buffer of object positions in their local FoR is traversed back in time in time steps of 50 ms. As long as the stimulus is in a given RF, *D* and *TTC* at every time step are recorded as positive examples on each taxel's data set.
2. For all other taxels of the same body part, the procedure is analogous, but negative examples are appended to their respective data sets.

Stimuli that move close to but never touch the body surface do not contribute to the peripersonal space representation. However, taking into account all events that come sufficiently close to the body surface would be an equally valid approach.

## Internal representation

Each taxel stores and continuously updates a record of the count of positive and negative examples that it has encountered for every combination of distance and time to contact. We defined the range of $D$ as $[-10, 20]$ cm and $TTC$ as $[0, 3]$ s. The variables were discretized into eight equally-sized bins for the distance and four bins for the time to contact respectively; the $TTC$ requires a velocity estimation of the approaching object and gives rise to noisier estimates. There are 32 combinations and hence 32 items, $[n_{positive}, n_{negative}]$, in every taxel's memory. As mentioned earlier, the main advantage of this representation is its simplicity and the possibility of incremental updates—for each new positive or negative example, the respective count in memory is incremented. However, most relevant for the robot is an estimation of the probability (density) of an object hitting a particular part of the skin, which can be used to trigger avoidance responses, for example. The stimulus's "coordinates" w.r.t. each taxel (i.e. distance, time to contact) can be discretized as described above and a frequentist probability estimate obtained simply as:

$$P(D, TTC) \approx f(D, TTC) = \frac{n_{positive}(D, TTC)}{n_{positive}(D, TTC) + n_{negative}(D, TTC)} \tag{3}$$

Such an approach—discretized representation and querying—constitutes the simplest solution. However, it may give rise to unstable performance, in particular in the case when the state space is undersampled. Therefore, it is desirable to obtain a continuous function $f$ which can be sampled at any real values of $[D, TTC]$. This can be achieved by using the Parzen-Window density estimation algorithm [61]—in fact, to interpolate the data. In a 1-dimensional case, the interpolated value $p(\mathbf{x})$ for any $\mathbf{x}$ is given by:

$$p(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{h^2} \Phi\left(\frac{\mathbf{x}_i - \mathbf{x}}{h}\right) \tag{4}$$

where $\mathbf{x}_i$ are the data points in the discrete input space, $\Phi$ is the window function or kernel and $h$ its bandwidth parameter, which is responsible for weighting the contributions of the neighbors of the point $\mathbf{x}$. We used a Gaussian function, hence we have:

$$p(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(\mathbf{x}_i - \mathbf{x})^2}{2\sigma^2}\right) \tag{5}$$

In our case, which is bi-dimensional (with $\mathbf{x} = [D, TTC]$ as the input variables), we specified the standard deviation $\sigma$ equal to the width of the single bin in each dimension of the input space. For any value of $D = d$ and $TTC = ttc$, the final interpolated value, $p(d, ttc)$, represents the probability of an object at distance $d$ and time to contact $ttc$ hitting the specific taxel under consideration. It is worth noting that only the original discretized $[D, TTC]$ combinations have estimates of a probability function associated with them, each pair $[D_i, TTC_j]$ independently from others. However, the whole "landscape" arising from $f(D, TTC)$ cannot be interpreted as a probability mass function (in discrete case) or probability density function (in continuous case), because the overall probability for the whole space of $D$ and $TTC$ combinations can take any values and does not sum to 1.

## Monte Carlo simulation of a single taxel

In order to investigate the quality of the representation proposed in Section Internal representation, a Monte Carlo simulation was implemented. In particular, we wanted to study the properties of the acquired representation in noiseless and noisy conditions—with sufficient samples

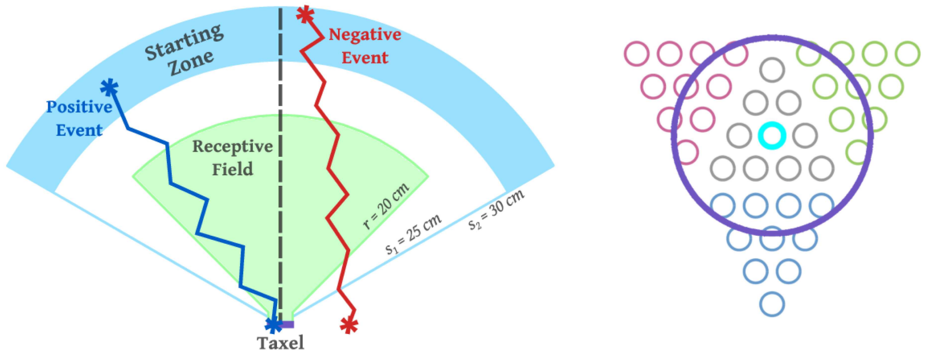


**Fig 15. 2D schematics of single taxel model. (Left)** Side view of the simulated taxel with examples of approaching stimuli: the purple slab at the bottom represents a taxel; the green sector is a projection of the taxel's cone-shaped receptive field; blue sector marks the region where stimuli are generated. The two examples show a positive event (blue line) and a negative one (red line). **(Right)** Top view of the simulated taxel with its nearby skin structure: the purple circle represents grouping of several sensors (physical taxels) in a modeled taxel (virtual).

doi:10.1371/journal.pone.0163713.g015

available and with control over noise—and investigate the effect of the hyper-parameters (such as number of bins for discretization, definition of the RF cone, range of stimuli's speed, etc.). To this end, a 3D model of a single taxel with simulated stimuli was prepared—see Fig 15 for an illustration of the simulation environment. The code with the complete model is available at the public repository [47].

The model parameters were chosen to mimic the real robot setup as closely as possible. The simulated taxel itself has a radius of $0.235cm$, which mimics the radius of the real iCub taxels. However, objects landing within $2cm$ from the taxel's center (purple areas in Fig 15) are still considered positive, resembling the size of a triangular module of the iCub skin which is itself composed of 10 taxels (see Fig 12 above). These "virtual taxels" will be used in the real setup by joining the responses of a number of adjacent physical taxels. The taxel's cone-shaped receptive field is depicted in green. Approaching stimuli are simulated by generating trajectories possibly corrupted by measurement noise. Since the nature of our data collection and learning method requires positive examples (objects contacting the virtual taxel) as well as negative examples (objects contacting neighboring taxels), we simulated three neighboring virtual taxels (Fig 15 right). We implemented a stochastic "shower" of objects with their starting points uniformly distributed in the blue region ("starting zone" in Fig 15 left) and their landing points following a Gaussian distribution centered on the simulated taxel ($\mu = 0$; $\sigma = 5cm$). The velocity of the object is a vector directed from the starting point to the landing point, whose speed is uniformly distributed between $5cm/s$ and $15cm/s$ (but constant over time for any given trajectory). Position and velocity are sampled with $T = 50ms$. Measurement noise is Gaussian both for position and velocity. The Monte Carlo simulation is implemented in Matlab.

## Avoidance and reaching controller

As an exploitation of the learned representations, we implemented a velocity controller that can move any point of either the left or right kinematic chain of the arms in a desired direction. During an avoidance task, the movement is directed away from the point of maximum activation, along the normal to the local surface in that point. For reaching, the desired movement vector has the opposite direction. We compute a weighted average for both the position of the avoidance/reaching behavior and its direction of motion as follows:

$$\begin{aligned}\mathbf{P}(t) &= \frac{1}{k}\sum_{i=1}^{k}[a_i(t)\cdot \mathbf{p}_i(t)] \\ \mathbf{N}(t) &= \frac{1}{k}\sum_{i=1}^{k}[a_i(t)\cdot \mathbf{n}_i(t)]\end{aligned} \tag{6}$$

where $\mathbf{P}(t)$ and $\mathbf{N}(t)$ are the desired position and direction of motion in the robot's Root FoR respectively, $\mathbf{p}_i(t)$ and $\mathbf{n}_i(t)$ are the individual taxels' positions and normals. These are weighted by the activations, $a_i(t)$, of the corresponding taxels' representations. The weighted average is computed by cycling through all the taxels whose activation is bigger than a predefined threshold at any given instant of time. Therefore, the resultant position and the direction of motion of the avoidance/reaching behavior are proportional to the activation of the taxels' representations and change dynamically as the activation levels of different taxels varies. The velocity control loop employs a Cartesian controller [57] whose reference speed was fixed to $10cm/s$.

## Supporting Information

**S1 Fig. End-effector trajectories in operational space during avoidance (red) and reaching (blue).** A schematic illustration of the robot's upper body kinematics during resting configuration is depicted. For each link—torso (gray), left arm (pink), right arm (light blue), right and left eye (yellow)—the end-effectors' reference frames are also shown.
(EPS)

**S2 Fig. Avoidance demonstration using distance only information. (Left)** Object approaching the inner part of left forearm. Top plot shows the distance of the object from the taxels in their individual FoRs. The shaded purple area marks the velocity of the body part (common to all taxels; maximum activation corresponding to $10cm/s$). Bottom plot depicts the activations of the forearm taxels' PPS representations. First approaching behavior was directed to the external part of the forearm (taxels in tones of green); second approach toward the internal part (taxels in tones of red) **(Right)** Object approaching the right palm. From [45].
(EPS)

## Author Contributions

**Conceptualization:** AR MH UP LF GM.

**Data curation:** AR MH.

**Formal analysis:** AR MH UP.

**Funding acquisition:** GM MH.

**Investigation:** AR MH UP.

**Methodology:** AR MH.

**Software:** AR MH.

**Supervision:** MH UP GM.

**Validation:** AR MH UP.

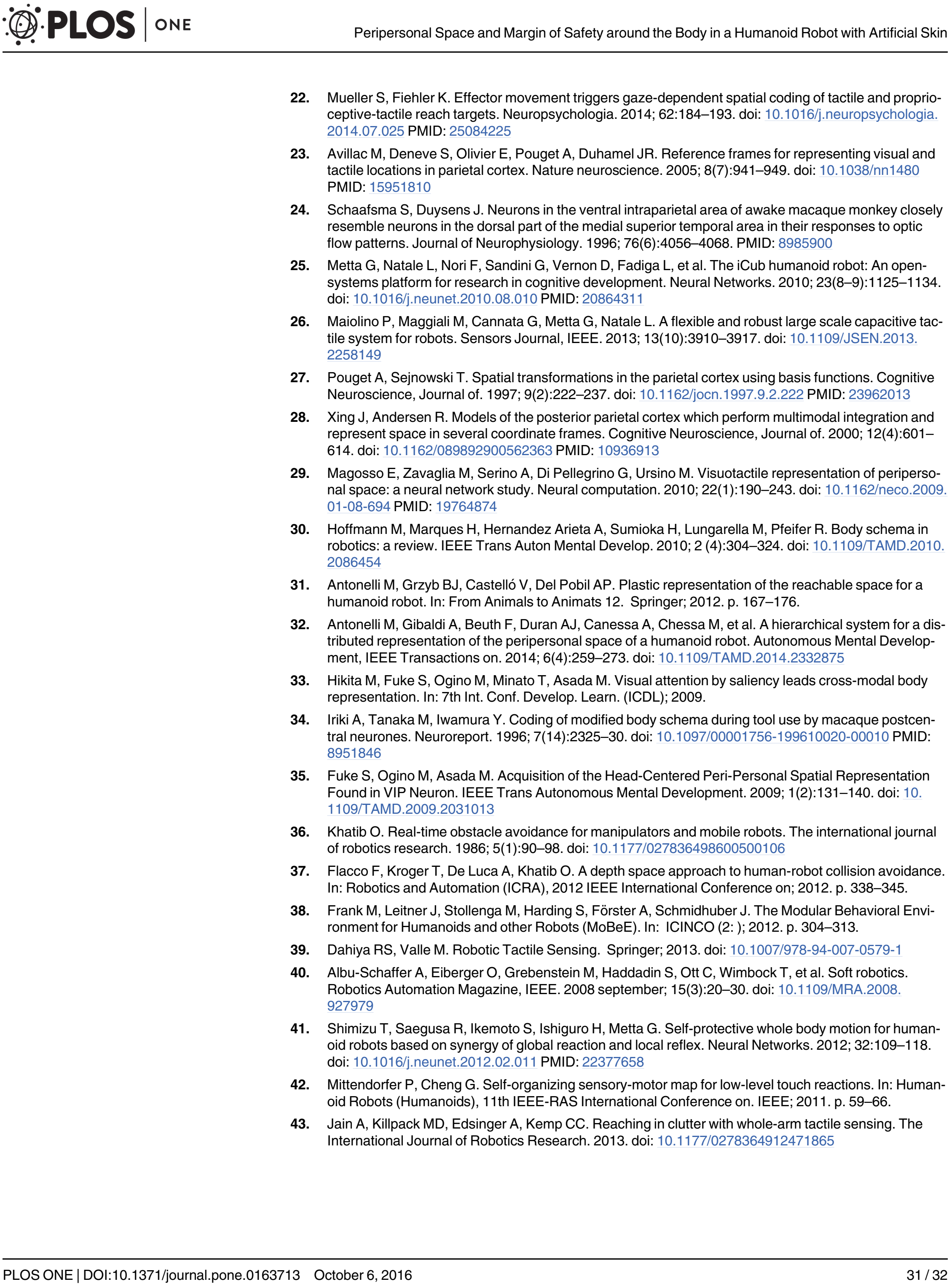
**Visualization:** AR MH.

**Writing – original draft:** AR MH.

**Writing – review & editing:** AR MH UP LF GM.

## References

1. Cléry J, Guipponi O, Wardak C, Hamed SB. Neuronal bases of peripersonal and extrapersonal spaces, their plasticity and their dynamics: knowns and unknowns. Neuropsychologia. 2015; 70:313–326. doi: 10.1016/j.neuropsychologia.2014.10.022 PMID: 25447371
2. Graziano MSA, Cooke DF. Parieto-frontal interactions, personal space and defensive behavior. Neuropsychologia. 2006; 44:845–859. doi: 10.1016/j.neuropsychologia.2005.09.009 PMID: 16277998
3. Holmes NP, Spence C. The body schema and the multisensory representation(s) of peripersonal space. Cogn Process. 2004 Jun; 5(2):94–105. doi: 10.1007/s10339-004-0013-3 PMID: 16467906
4. Gentilucci M, Fogassi L, Luppino G, Matelli M, Camarda R, Rizzolatti G. Functional organization of inferior area 6 in the macaque monkey. Experimental brain research. 1988; 71(3):475–490. doi: 10.1007/BF00248741 PMID: 3416964
5. Fogassi L, Gallese V, Fadiga L, Luppino G, Matelli M, Rizzolatti G. Coding of peripersonal space in inferior premotor cortex (area F4). Journal of Neurophysiology. 1996; 76(1):141–157. PMID: 8836215
6. Rizzolatti G, Fadiga L, Fogassi L, Gallese V. The Space Around Us. Science. 1997; 277(5323):190–191. doi: 10.1126/science.277.5323.190 PMID: 9235632
7. Graziano MSA, Gandhi S. Location of the polysensory zone in the precentral gyrus of anesthetized monkeys. Experimental Brain Research. 2000; 135(2):259–266. doi: 10.1007/s002210000518 PMID: 11131511
8. Colby CL, Duhamel JR, Goldberg ME. Ventral intraparietal area of the macaque: anatomic location and visual response properties. Journal of Neurophysiology. 1993; 69(3):902–914. PMID: 8385201
9. Graziano MSA, Reiss LAJ, Gross CG. A neuronal representation of the location of nearby sounds. Nature. 1999; 397(6718):428–430. doi: 10.1038/17115 PMID: 9989407
10. Bremmer F, Schlack A, Shah NJ, Zafiris O, Kubischik M, Hoffmann KP, et al. Polymodal motion processing in posterior parietal and premotor cortex: a human fMRI study strongly implies equivalencies between humans and monkeys. Neuron. 2001; 29(1):287–296. doi: 10.1016/S0896-6273(01)00198-2 PMID: 11182099
11. Rochat P. Self-perception and action in infancy. Exp Brain Res. 1998; 123:102–109. doi: 10.1007/s002210050550 PMID: 9835398
12. Fukson OI, Berkinblit MB, Feldman AG. The spinal frog takes into account the scheme of its body during the wiping reflex. Science. 1980; 209:1261–1263. doi: 10.1126/science.7403886 PMID: 7403886
13. Berkinblit MB, Feldman AG, Fukson OI. Adaptability of innate motor patterns and motor control mechanisms. Behavioral and Brain Sciences. 1986; 9:585–599. doi: 10.1017/S0140525X00051268
14. MacKay-Lyons M. Central pattern generation of locomotion: a review of the evidence. Physical Therapy. 2002; 82:69–83. PMID: 11784280
15. Hulse M, McBride S, Law J, Lee M. Integration of active vision and reaching from a developmental robotics perspective. Autonomous Mental Development, IEEE Transactions on. 2010; 2(4):355–367. doi: 10.1109/TAMD.2010.2081667
16. Piaget J. The construction of reality in the child. New York: Basic Books; 1954. doi: 10.1007/s10071-014-0788-2 PMID: 25092491
17. Thomas BL, Karl JM, Whishaw IQ. Independent development of the Reach and the Grasp in spontaneous self-touching by human infants in the first 6 months. Frontiers in psychology. 2015; 5. doi: 10.3389/fpsyg.2014.01526 PMID: 25620939
18. Roncone A, Hoffmann M, Pattacini U, Metta G. Automatic kinematic chain calibration using artificial skin: self-touch in the iCub humanoid robot. In: Proc. IEEE Int. Conf. Robotics and Automation (ICRA); 2014. p. 2305–2312.
19. Heed T, Buchholz VN, Engel AK, Röder B. Tactile remapping: from coordinate transformation to integration in sensorimotor processing. Trends in cognitive sciences. 2015; 19(5):251–258. doi: 10.1016/j.tics.2015.03.001 PMID: 25843541
20. Roncone A, Pattacini U, Metta G, Natale L. A Cartesian 6-DoF Gaze Controller for Humanoid Robots. In: Proceedings of Robotics: Science and Systems. AnnArbor, Michigan; 2016.
21. Harrar V, Harris LR. Touch used to guide action is partially coded in a visual reference frame. Experimental brain research. 2010; 203(3):615–620. doi: 10.1007/s00221-010-2252-0 PMID: 20428854



22. Mueller S, Fiehler K. Effector movement triggers gaze-dependent spatial coding of tactile and proprioceptive-tactile reach targets. Neuropsychologia. 2014; 62:184–193. doi: 10.1016/j.neuropsychologia.2014.07.025 PMID: 25084225

23. Avillac M, Deneve S, Olivier E, Pouget A, Duhamel JR. Reference frames for representing visual and tactile locations in parietal cortex. Nature neuroscience. 2005; 8(7):941–949. doi: 10.1038/nn1480 PMID: 15951810

24. Schaafsma S, Duysens J. Neurons in the ventral intraparietal area of awake macaque monkey closely resemble neurons in the dorsal part of the medial superior temporal area in their responses to optic flow patterns. Journal of Neurophysiology. 1996; 76(6):4056–4068. PMID: 8985900

25. Metta G, Natale L, Nori F, Sandini G, Vernon D, Fadiga L, et al. The iCub humanoid robot: An open-systems platform for research in cognitive development. Neural Networks. 2010; 23(8–9):1125–1134. doi: 10.1016/j.neunet.2010.08.010 PMID: 20864311

26. Maiolino P, Maggiali M, Cannata G, Metta G, Natale L. A flexible and robust large scale capacitive tactile system for robots. Sensors Journal, IEEE. 2013; 13(10):3910–3917. doi: 10.1109/JSEN.2013.2258149

27. Pouget A, Sejnowski T. Spatial transformations in the parietal cortex using basis functions. Cognitive Neuroscience, Journal of. 1997; 9(2):222–237. doi: 10.1162/jocn.1997.9.2.222 PMID: 23962013

28. Xing J, Andersen R. Models of the posterior parietal cortex which perform multimodal integration and represent space in several coordinate frames. Cognitive Neuroscience, Journal of. 2000; 12(4):601–614. doi: 10.1162/089892900562363 PMID: 10936913

29. Magosso E, Zavaglia M, Serino A, Di Pellegrino G, Ursino M. Visuotactile representation of peripersonal space: a neural network study. Neural computation. 2010; 22(1):190–243. doi: 10.1162/neco.2009.01-08-694 PMID: 19764874

30. Hoffmann M, Marques H, Hernandez Arieta A, Sumioka H, Lungarella M, Pfeifer R. Body schema in robotics: a review. IEEE Trans Auton Mental Develop. 2010; 2 (4):304–324. doi: 10.1109/TAMD.2010.2086454

31. Antonelli M, Grzyb BJ, Castelló V, Del Pobil AP. Plastic representation of the reachable space for a humanoid robot. In: From Animals to Animats 12. Springer; 2012. p. 167–176.

32. Antonelli M, Gibaldi A, Beuth F, Duran AJ, Canessa A, Chessa M, et al. A hierarchical system for a distributed representation of the peripersonal space of a humanoid robot. Autonomous Mental Development, IEEE Transactions on. 2014; 6(4):259–273. doi: 10.1109/TAMD.2014.2332875

33. Hikita M, Fuke S, Ogino M, Minato T, Asada M. Visual attention by saliency leads cross-modal body representation. In: 7th Int. Conf. Develop. Learn. (ICDL); 2009.

34. Iriki A, Tanaka M, Iwamura Y. Coding of modified body schema during tool use by macaque postcentral neurones. Neuroreport. 1996; 7(14):2325–30. doi: 10.1097/00001756-199610020-00010 PMID: 8951846

35. Fuke S, Ogino M, Asada M. Acquisition of the Head-Centered Peri-Personal Spatial Representation Found in VIP Neuron. IEEE Trans Autonomous Mental Development. 2009; 1(2):131–140. doi: 10.1109/TAMD.2009.2031013

36. Khatib O. Real-time obstacle avoidance for manipulators and mobile robots. The international journal of robotics research. 1986; 5(1):90–98. doi: 10.1177/027836498600500106

37. Flacco F, Kroger T, De Luca A, Khatib O. A depth space approach to human-robot collision avoidance. In: Robotics and Automation (ICRA), 2012 IEEE International Conference on; 2012. p. 338–345.

38. Frank M, Leitner J, Stollenga M, Harding S, Förster A, Schmidhuber J. The Modular Behavioral Environment for Humanoids and other Robots (MoBeE). In: ICINCO (2: ); 2012. p. 304–313.

39. Dahiya RS, Valle M. Robotic Tactile Sensing. Springer; 2013. doi: 10.1007/978-94-007-0579-1

40. Albu-Schaffer A, Eiberger O, Grebenstein M, Haddadin S, Ott C, Wimbock T, et al. Soft robotics. Robotics Automation Magazine, IEEE. 2008 september; 15(3):20–30. doi: 10.1109/MRA.2008.927979

41. Shimizu T, Saegusa R, Ikemoto S, Ishiguro H, Metta G. Self-protective whole body motion for humanoid robots based on synergy of global reaction and local reflex. Neural Networks. 2012; 32:109–118. doi: 10.1016/j.neunet.2012.02.011 PMID: 22377658

42. Mittendorfer P, Cheng G. Self-organizing sensory-motor map for low-level touch reactions. In: Humanoid Robots (Humanoids), 11th IEEE-RAS International Conference on. IEEE; 2011. p. 59–66.

43. Jain A, Killpack MD, Edsinger A, Kemp CC. Reaching in clutter with whole-arm tactile sensing. The International Journal of Robotics Research. 2013. doi: 10.1177/0278364912471865

44. De Luca A, Flacco F. Integrated control for pHRI: Collision avoidance, detection, reaction and collaboration. In: Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEEE RAS EMBS International Conference on; 2012. p. 288–295.

45. Roncone A, Hoffmann M, Pattacini U, Metta G. Learning peripersonal space representation through artificial skin for avoidance and reaching with whole body surface. In: Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on; 2015. p. 3366–3373.

46. Roncone A, Hoffmann M, Pattacini U, Fadiga L, Metta G. Supporting material—source code; 2016. Available from: https://github.com/robotology/peripersonal-space.

47. Roncone A, Hoffmann M, Pattacini U, Fadiga L, Metta G. Supporting material—data and scripts; 2016. Available from: https://github.com/alecive/peripersonal-space-margin-of-safety-data.

48. Del Prete A, Denei S, Natale L, M F, Nori F, Cannata G, et al. Skin spatial calibration using force/torque measurements. In: IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS); 2011. p. 3694 –3700.

49. Pattacini U. Modular cartesian controllers for humanoid robots: Design and implementation on the iCub. Ph. D. dissertation, RBCS, Italian Institute of Technology, Genova; 2011.

50. Averbeck BB, Latham PE, Pouget A. Neural correlations, population coding and computation. Nature Reviews Neuroscience. 2006; 7(5):358–366. doi: 10.1038/nrn1888 PMID: 16760916

51. Corbetta D, Thurman SL, Wiener RF, Guan Y, Williams JL. Mapping the feel of the arm with the sight of the object: on the embodied origins of infant reaching. Frontiers in psychology. 2014; 5:576. doi: 10.3389/fpsyg.2014.00576 PMID: 24966847

52. Sclafani V, Simpson EA, Suomi SJ, Ferrari PF. Development of space perception in relation to the maturation of the motor system in infant rhesus macaques (Macaca mulatta). Neuropsychologia. 2014. doi: 10.1016/j.neuropsychologia.2014.12.002 PMID: 25486636

53. Rizzolatti G, Fogassi L, Gallese V. Motor and cognitive functions of the ventral premotor cortex. Current opinion in neurobiology. 2002; 12(2):149–154. doi: 10.1016/S0959-4388(02)00308-2 PMID: 12015230

54. Cohen YE, Andersen RA. A common reference frame for movement plans in the posterior parietal cortex. Nature Reviews Neuroscience. 2002; 3(7):553–562. doi: 10.1038/nrn873 PMID: 12094211

55. Parmiggiani A, Maggiali M, Natale L, Nori F, Schmitz A, Tsagarakis N, et al. The design of the iCub humanoid robot. International Journal of Humanoid Robotics. 2012; 9(04). doi: 10.1142/S0219843612500272

56. Bradski G. The OpenCV Library. Dr Dobb's Journal of Software Tools. 2000.

57. Pattacini U, Nori F, Natale L, Metta G, Sandini G. An experimental evaluation of a novel minimum-jerk Cartesian controller for humanoid robots. In: Proc. IEEE/RSJ Int. Conf. Int. Robots and Systems (IROS); 2010.

58. Ciliberto C, Pattacini U, Natale L, Nori F, Metta G. Reexamining Lucas-Kanade method for real-time independent motion detection: application to the iCub humanoid robot. In: Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on. IEEE; 2011. p. 4154–4160.

59. Tikhanoff V, Pattacini U, Natale L, Metta G. Exploring affordances and tool use on the iCub. In: Humanoid Robots (Humanoids), 13th IEEE-RAS International Conference on. IEEE; 2013.

60. Fanello SR, Pattacini U, Gori I, Tikhanoff V, Randazzo M, Roncone A, et al. 3D Stereo Estimation and Fully Automated Learning of Eye-Hand Coordination in Humanoid Robots. In: Humanoid Robots (Humanoids), 14th IEEE-RAS International Conference on; 2014.

61. Parzen E. On Estimation of a Probability Density Function and Mode. The Annals of Mathematical Statistics. 1962; 33(3):1065–1076. doi: 10.1214/aoms/1177704472

# Appendix H

# Robot self-calibration using multiple kinematic chains – a simulation study on the iCub humanoid robot

Authors' version of:

Stepanova, K. and Pajdla, T., and Hoffmann, M. (2019). Robot self-calibration using multiple kinematic chains – a simulation study on the iCub humanoid robot. *IEEE Robotics and Automation Letters* 4 (2): 1900-1907.

DOI: `https://doi.org/10.1109/LRA.2019.2898320`
Youtube video: `https://youtu.be/zP3c7Eq8yVk`

Author contributions: The contribution of M. Hoffmann was 40%.

IEEE ROBOTICS AND AUTOMATION LETTERS. PREPRINT VERSION. ACCEPTED JANUARY, 2019 1

# Robot self-calibration using multiple kinematic chains – a simulation study on the iCub humanoid robot

Karla Stepanova[1,2], Tomas Pajdla[2], and Matej Hoffmann[1]

***Abstract*—Mechanism calibration is an important and non-trivial task in robotics. Advances in sensor technology make affordable but increasingly accurate devices such as cameras and tactile sensors available, making it possible to perform automated self-contained calibration relying on redundant information in these sensory streams. In this work, we use a simulated iCub humanoid robot with a stereo camera system and end-effector contact emulation to quantitatively compare the performance of kinematic calibration by employing different combinations of intersecting kinematic chains—either through self-observation or self-touch. The parameters varied were: (i) type and number of intersecting kinematic chains used for calibration, (ii) parameters and chains subject to optimization, (iii) amount of initial perturbation of kinematic parameters, (iv) number of poses/configurations used for optimization, (v) amount of measurement noise in end-effector positions / cameras. The main findings are: (1) calibrating parameters of a single chain (e.g. one arm) by employing multiple kinematic chains ("self-observation" and "self-touch") is superior in terms of optimization results as well as observability; (2) when using multi-chain calibration, fewer poses suffice to get similar performance compared to when for example only observation from a single camera is used; (3) parameters of all chains (here 86 DH parameters) can be subject to calibration simultaneously and with 50 (100) poses, end-effector error of around 2 (1) mm can be achieved; (4) adding noise to a sensory modality degrades performance of all calibrations employing the chains relying on this information.**

***Index Terms*—Humanoid robots, calibration and identification, force and tactile sensing, kinematics, optimization and optimal control.**

## I. INTRODUCTION

ROBOTS performing manipulation tasks rely on models of their bodies and their success is largely determined by their accuracy. However, inaccuracies creep in many ways as for example in the assembly process, in mechanical elasticity, or simply because of cheap design of components. Therefore, the actual model parameters of every robot exemplar have to be found by means of a calibration procedure, usually relying on external metrology systems. For kinematic calibration, such apparatuses can measure one or more of the components of the end-effector pose employing mechanical, visual, or laser systems (see [1] for a survey). Different arrangements have different accuracy, requirements on the environment, and cost. These conditions have to be present for recalibration to be performed.

Current trends in the robotics industry make classical calibration procedures less practical: with the advent of the so-called "collaborative robots", for example, the machines are becoming cheaper, lightweight, compliant, and they are being deployed in more versatile ways according to the needs of customized production of smaller batches rather than being fixed in a single production line for their entire lifetime. All these factors increase the need for calibration to be performed more frequently. At the same time, the machines, including home and service robots, often come with richer sets of powerful sensory devices that are affordable and not difficult to operate. Both these trends speak for alternative solutions to the self-calibration problem that are more "self-contained" and can be performed autonomously by the robot.

Hollerbach et al. [1] classify different calibration methods into *open-loop*—where one or more of the components of the end-effector pose is measured employing mechanical, visual, or laser systems—and *closed-loop* where physical constraints on the end-effector position or orientation can substitute for measurements. Observing the end-effector—or in general any other points on the kinematic chain—using a camera falls into the open-loop calibration family, although components of the end-effector pose can be observed only indirectly through projection into the camera frame. Self-touch configurations employing two arms of the humanoid robot could be framed as a constraint if contact measurement only (e.g. from force/torque sensors) was available and hence treated as closed-loop. In this work, we follow up on [2] and emulate sensitive skin measurements, which provide the position of contact (and hence fit more naturally with open-loop calibration).

Our work is a simulation study that draws on calibration in the real world—like different approaches to kinematic calibration of the iCub humanoid robot relying on self-observation [3], [4] and self-touch [2]. Using the model of the robot with identical parameters, but exploiting the fact that we have complete knowledge of the system and capacity to emulate

Manuscript received: September 10, 2018; Revised: December 11, 2018; Accepted: January 16, 2019

This paper was recommended for publication by Editor Allison Okamura upon evaluation of the Associate Editor Nikos Tsagarakis and Reviewers' comments. K.S. and M.H. were supported by the Czech Science Foundation under Project GA17-15697Y; T.P. was supported by the European Regional Development Fund under the project Robotics for Industry 4.0 (reg. no. CZ.02.1.01/0.0/0.0/15 003/0000470)

[1] Karla Stepanova and Matej Hoffmann are with Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic. karla.stepanova@cvut.cz and matej.hoffmann@fel.cvut.cz

[2] Karla Stepanova and Tomas Pajdla are with Czech Institute of Informatics, Robotics, and Cybernetics, Czech Technical University in Prague, Czech Republic pajdla@cvut.cz

Digital Object Identifier (DOI): see top of this page.

different levels of model perturbation and measurement noise, our goal is to get insights into the pros and cons of different optimization problem formulations. In particular, we study how the calibration performance is dependent on the type and number of intersecting kinematic chains, the number of parameters calibrated, number of robot configurations, and the measurement noise. Accompanying video is available here https://youtu.be/zP3c7Eq8yVk and dataset at [5].

This article is structured as follows. Related work is reviewed in the next section, followed by Materials and Methods, Data Acquisition and Description, and Simulation Results. We close with a Discussion and Conclusion.

## II. Related Work

We focus on humanoid robots or humanoid-like setups with many Degrees of Freedom (DoF) of two arms that can possibly self-touch, equipped with cameras and tactile or inertial sensors. These are challenging setups for calibration but they create new opportunities for automated self-contained calibration based on closing kinematic loops by touch (self-contact) and vision.

Most often, the loops are closed through self-observation of the end-effector using cameras located in the robot head (*open-loop calibration* method per [1]). Hersch et al. [6] and Martinez-Cantin et al. [7] present online methods to calibrate humanoid torso kinematics relying on gradient descent and recursive least squares estimation, respectively. The iCub humanoid was employed in [3], [4]. Vicente et al. [4] used a model of the hand's appearance to estimate its 6D pose and used that information to calibrate the joint offsets. Fanello et al. [3] had the robot observe its fingertip and learned essentially a single transformation only to account for the discrepancy between forward kinematics of the arm and the projection of the finger into the cameras.

Next to cameras, inertial sensors also contain information that can be exploited for calibration. Kinematic calibration was shown exploiting 3-axis accelerometers embedded in the artificial skin modules distributed on robot body [8], [9] or in the control boards on the iCub [10] or CMU/Sarcos [11].

The advent of robotic skin technologies [12], [13] opens up the possibility of a new family of approaches, whereby the chain is closed through contact like in closed-loop calibration, but the contact position can be extracted from the tactile array. Roncone et al. [2] showed this on the iCub robot that performs autonomous self-touch using a finger with sensitive fingertip to touch the skin-equipped forearm of the contralateral arm; Li et al. [14] employed a dual KUKA arm setup with a sensorized "finger" and a tactile array on the other manipulator. Forward kinematics together with skin calibration provide contact position that can then be used for robot kinematic calibration. In this sense, the skin provides a pose measurement rather than constraint and as such, this may fall under *open-loop calibration*. In this way, one arm of a humanoid can be used to calibrate the other. Khusainov et al. [15] exploit this principle using an industrial manipulator to calibrate the legs of a humanoid robot. Another variant is exploiting the sensitive fingertips to touch a known external surface [16].

Birbach et al. [17] were to our knowledge the only ones to employ truly "multisensorial" or "multimodal" calibration. Using the humanoid robot Justin observing its wrist, the error functions comparing the wrist's position from forward kinematics with its projection into the left and right camera images, Kinect image, and Kinect disparity, together with an inertial term, were aggregated into a single cost function to be minimized. It is claimed that while pair-wise calibration can lead to inconsistencies, calibrating everything together in a "mutually supportive way" is most efficient.

In this work, we compare calibration through self-observation (with projection into cameras) and calibration through self-touch and the effect of their synergy. Our work makes a unique contribution, also compared to [17] who, first, employ essentially only "hand-eye" kinematic chains terminating in different vision-like sensors in the robot head, and, second, consider only the case where all chains are combined together using a single cost function.

## III. Materials and Methods

### A. *iCub robot kinematic model and camera parameters*

In this work, we use the upper body of the iCub humanoid robot (see Fig. 1) and its kinematic model expressed in the Denavit-Hartenberg convention, where every link $i$ is described by 4 parameters: $\{a_i, d_i, \alpha_i, o_i\}$. In this platform, all joints are revolute. We will consider several kinematic chains: all start in a single inertial or base frame—denoted iCub *Root* Reference Frame here. For every chain, the DH parameters uniquely define a chain of transformation matrices from the inertial frame to the end-effector. The position and orientation of the end-effector in the *Root* frame is thus given by $\boldsymbol{T}_n^{Root} = A_1(q_1)...A_n(q_n)$ where the homogeneous transformation matrices $A_i$ can be constructed from the DH representation and $q_i$ are current joint angles of the robot actuators.

The links are schematically illustrated in Fig. 1. iCub kinematics version 1 was used [18] with the following modification: the *Root* was moved from the waist area to the third torso joint, which is the new inertial frame for our purposes.

The four chains under consideration are:

1) Left Arm (LA). DH parameters in Table I. Short names to denote the links/joints: Root-to-LAshoulder, LA Shoulder Pitch, LA Shoulder Roll, LA Shoulder Yaw, LA Elbow, LA Wrist Prosup (for pronosupination), LA Wrist Pitch, LA Wrist Yaw.
2) Right Arm (RA). DH parameters analogous to LA (see [18]). Link/joint names: Root-to-RAshoulder, RA Shoulder Pitch, RA Shoulder Roll, RA Shoulder Yaw, RA Elbow, RA Wrist Prosup, RA Wrist Pitch, RA Wrist Yaw.
3) Left Eye (LEye). DH parameters in Table II. Link/joint names: Root-to-neck, Neck Pitch, Neck Roll, Neck Yaw, Eyes Tilt, Left Eye Pan.
4) Right Eye (REye). DH parameters different than LEye in Table III. Link/joint names: Root-to-neck, Neck Pitch, Neck Roll, Neck Yaw, Eyes Tilt, Right Eye Pan.

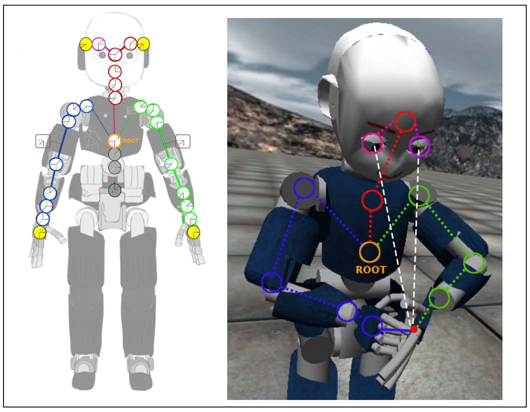


Fig. 1. iCub upper body and schematic illustration of kinematic chains considered. All chains originate in a common *Root* which is located at the third torso joint. The left and right arm chains are drawn in green and blue respectively. The eye chains have a common Root-to-head chain part marked in red. The right panel illustrates the self-calibration by connecting different chains—self-touch and self-observation. White lines denote projection into the eyes/cameras.

Links or parameters not subject to calibration are showed shaded in grey in the corresponding tables. The first link always originates in the Root frame and is fixed in all chains (the torso joint is not moving) and is also excluded from calibration. The alpha parameter of the last link in the arm chains is also not being calibrated as it is not observable because we observe only position and not the orientation of the end-effectors. The right arm chain is further extended with a fixed transform from the end-effector in the palm to the tip of the index finger—not subject to calibration. The eye chains differ in the last link only.

| Link(i) | a(i) [mm] | d(i) [mm] | $\alpha$ [rad] | $o$ [rad] |
|---|---|---|---|---|
| 1 | 23.36 | 143.3 | $\pi/2$ | $105 * \pi/180$ |
| 2 | 0 | 107.74 | $-\pi/2$ | $\pi/2$ |
| 3 | 0 | 0 | $\pi/2$ | $-\pi/2$ |
| 4 | 15 | 152.28 | $-\pi/2$ | $75 * \pi/180$ |
| 5 | -15 | 0 | $\pi/2$ | 0 |
| 6 | 0 | 137.3 | $\pi/2$ | $-\pi/2$ |
| 7 | 0 | 0 | $\pi/2$ | $\pi/2$ |
| 8 | 62.5 | -16 | 0 | 0 |

TABLE I
DH PARAMETERS ($a, d, \alpha$ AND OFFSETS $o$) DESCRIBING ALL LINKS IN LEFT ARM KINEMATIC CHAIN.

| Link(i) | a(i) [mm] | d(i) [mm] | $\alpha$ [rad] | $o$ [rad] |
|---|---|---|---|---|
| 1 | 2.31 | - 193.3 | $-\pi/2$ | $\pi/4$ |
| 2 | 33 | 0 | $\pi/2$ | $\pi/4$ |
| 3 | 0 | 1 | $-\pi/2$ | $\pi/4$ |
| 4 | - 54 | 82.5 | $-\pi/2$ | $\pi/4$ |
| 5 | 0 | - 34 | $-\pi/2$ | 0 |
| 6 | 0 | 0 | $\pi/2$ | $-\pi/4$ |

TABLE II
DH PARAMETERS – LEFT EYE KINEMATIC CHAIN.

The camera intrinsic parameters were taken from the real robot cameras and were not subject to calibration: resolution 320 x 240, focal length $f_x = 257.34$, $f_y = 257.34.c_y = 120$.

| Link(i) | a(i) [mm] | d(i) [mm] | $\alpha$ [rad] | $o$ [rad] |
|---|---|---|---|---|
| 5 | 0 | 34 | $\pi/2$ | $-\pi/4$ |
| 6 | 0 | 0 | $-\pi/2$ | 0 |

TABLE III
DH PARAMETERS – RIGHT EYE KINEMATIC CHAIN. LINKS 1-4 SHARED WITH LEFT EYE KINEMATIC CHAIN.

### B. Optimization problem formulation

By calibration we mean estimation of the parameter vector $\boldsymbol{\phi} = \{[a_1, ..., a_n], [d_1, ..., d_n], [\alpha_1, ..., \alpha_n], [o_1, ..., o_n]\}$ with $i \in N$, where $N = \{1, .., n\}$ is a set of indices identifying individual links; $a$, $d$ and $\alpha$ are the first three parameters of the DH formulation and $o$ the offset that specifies the positioning of the encoders on the joints with respect to the DH representation. We often estimate a subset of these parameters only, assuming that the others are known. This subset can for example consist of a subset of links $N' \subset N$ (e.g., only parameters of one arm are to be calibrated) or a subset of the parameters (e.g., only offsets $o$ are to be calibrated—sometimes dubbed "daily calibration" [19]).

The estimation of the parameter vector $\boldsymbol{\phi}$ is done by optimizing a given objective function:

$$\boldsymbol{\phi}^* = \underset{\boldsymbol{\phi}}{\operatorname{argmin}} \sum_{m=1}^{M} ||\boldsymbol{p}_m^r - \boldsymbol{p}_m^e(\boldsymbol{\phi}, \boldsymbol{\Theta}_m)||^2, \qquad (1)$$

where $M$ is the number of robot configurations and corresponding end-effector positions used for calibration (hereafter, often referred to as "poses" for short), $\boldsymbol{p}_m^r$ is a real (observed) end-effector position, $\boldsymbol{p}_m^e$ is an estimated end-effector position computed using forward kinematic function for a given parameter estimate $\boldsymbol{\phi}$ and joint angles from joint encoders $\boldsymbol{\Theta}_m$. For chains involving cameras, the reprojection error is used instead, as described in the next section.

### C. Kinematic chain calibration

We study different combinations of intersecting chains and their performance in calibrating one another.

*1) Two arms chain (LA-RA):* This corresponds to the self-touch scenario, with touch occurring directly at the end-effectors (the right arm end-effector being shifted from palm to tip of index finger using a fixed transform). The newly established kinematic chain for upper body includes both arms while head and eyes are excluded. To optimize parameters describing this chain, we minimize the distance between estimated positions in 3D space of left and right arm end-effectors. In this case, the parameter vector $\boldsymbol{\phi}$ consists of the following parameters: $\boldsymbol{\phi} = \{\boldsymbol{\phi}^r, \boldsymbol{\phi}^l\}$, where $\boldsymbol{\phi}^r$ and $\boldsymbol{\phi}^l$ are parameters corresponding to the robot right and left arm, respectively. The objective function to be optimized is:

$$\boldsymbol{\phi}^* = \underset{\boldsymbol{\phi}}{\operatorname{argmin}} \sum_{m=1}^{M} ||\boldsymbol{X}_m^{r,R}(\boldsymbol{\phi}^r, \boldsymbol{\Theta}_m^r) - \boldsymbol{X}_m^{l,R}(\boldsymbol{\phi}^l, \boldsymbol{\Theta}_m^l)||^2 \quad (2)$$

where $M$ is the number of poses used for calibration, $\boldsymbol{X}_m^{r,R}$ and $\boldsymbol{X}_m^{l,R}$ are the $m^{\text{th}}$ estimated end-effector positions in the Root frame for the right and left arm respectively, computed using a given parameter estimate $\boldsymbol{\phi}$ and joint angles from joint encoders $\boldsymbol{\Theta}_m$.

*2) Hand to eye chains (LA-LEye, LA-REye, RA-LEye, RA-REye):* To predict position of the end-effector in each of the robot cameras (similar to [17]), the estimated end-effector position, $\boldsymbol{X}^{Root}$, is given by a current hypothetical robot calibration of the parameter vector $\boldsymbol{\phi}$ and is computed via forward kinematics. $\boldsymbol{X}^{Root}$ is then mapped to left camera coordinates ($\boldsymbol{X}^{LEye}$) using a transformation matrix $\boldsymbol{T}^{LEye}_{Root}$. Then we use a pinhole camera model to transform the 3D point ($\boldsymbol{X}^{LEye}$) into image coordinates ($\boldsymbol{X}^{img}$):

$$\begin{pmatrix} X^{img}_x \\ X^{img}_y \end{pmatrix} = \begin{pmatrix} f_x X^{LEye}_x / X^{LEye}_z \\ f_y X^{LEye}_y / X^{LEye}_z \end{pmatrix}, \tag{3}$$

where $f_x$, $f_y$ are focal lengths of the camera. Radial distortion of cameras was not considered.

This approach doesn't require information from both eyes and enables us to estimate only one side of the robot body (e.g. parameters of the left arm and left eye). For example, the estimated parameter vector $\boldsymbol{\phi}$ in the case of the kinematic chain connecting left arm and left eye consists of the following parameters: $\boldsymbol{\phi} = \{\boldsymbol{\phi}^l, \boldsymbol{\phi}^{le}\}$, where $\boldsymbol{\phi}^l$ and $\boldsymbol{\phi}^{le}$ are parameters corresponding to the robot left arm and to the left eye, respectively. The objective function is then defined as:

$$\boldsymbol{\phi}^* = \underset{\boldsymbol{\phi}}{\operatorname{argmin}} \sum_{m=1}^{M} ||\boldsymbol{X}^{l,img}_m(\boldsymbol{\phi}^l, \boldsymbol{\phi}^{le}) - \boldsymbol{u}^L_m||^2, \tag{4}$$

where $\boldsymbol{X}^{l,img}_m$ is the $m^{\text{th}}$ 2D position of the estimated left arm end-effector projected to left eye image coordinates and $\boldsymbol{u}^L_m$ is the $m$th 2D position of the observed left arm end-effector in the left camera. For two arms and two eyes we get four possible combined chains: left/right arm to right/left eye. Since the results are similar due to symmetry, we present in the experimental section results only for the Left arm - Left eye (LA-LEye) chain.

*3) Combining multiple chains (LA-RA-LEye, LA-RA-LEye-REye):* In order to estimate all kinematic parameters of the robot, we can take advantage of combining some or all of the above mentioned kinematic chains. For example, in the case that we combine LA-RA, LA-LEye and LA-REye chains together into LA-RA-LReye, the estimated parameter vector $\boldsymbol{\phi}$ consists of the following parameters: $\boldsymbol{\phi} = \{\boldsymbol{\phi}^r, \boldsymbol{\phi}^l, \boldsymbol{\phi}^{re}, \boldsymbol{\phi}^{le}\}$, where $\boldsymbol{\phi}^l$, $\boldsymbol{\phi}^r$, $\boldsymbol{\phi}^{re}$, and $\boldsymbol{\phi}^{le}$ are parameters corresponding to the left arm, right arm, right eye, and left eye, respectively. The objective function is in this case defined as:

$$\begin{aligned} \boldsymbol{\phi}^* = \underset{\boldsymbol{\phi}}{\operatorname{argmin}} \sum_{m=1}^{M} \{ & \mu \cdot ||\boldsymbol{X}^{r,R}_m(\boldsymbol{\phi}^r, \boldsymbol{\Theta}^r_m) - \boldsymbol{X}^{l,R}_m(\boldsymbol{\phi}^l, \boldsymbol{\Theta}^l_m)||+ \\ & ||\boldsymbol{X}^{l_L,I}_m(\boldsymbol{\phi}^l, \boldsymbol{\phi}^{le}) - \boldsymbol{u}^{l_L}_m|| + ||\boldsymbol{X}^{r_L,I}_m(\boldsymbol{\phi}^r, \boldsymbol{\phi}^{le}) - \boldsymbol{u}^{r_L}_m||+ \\ & ||\boldsymbol{X}^{l_R,I}_m(\boldsymbol{\phi}^l, \boldsymbol{\phi}^{re}) - \boldsymbol{u}^{l_R}_m|| + ||\boldsymbol{X}^{r_R,I}_m(\boldsymbol{\phi}^r, \boldsymbol{\phi}^{re}) - \boldsymbol{u}^{r_R}_m||\}^2, \end{aligned} \tag{5}$$

where $M$ is the number of poses (configurations) used for calibration, $\boldsymbol{X}^{r,R}_m$ and $\boldsymbol{X}^{l,R}_m$ are the $m^{\text{th}}$ estimated end-effector positions in the Root frame for the right and left arm, respectively. These are computed using a given parameter estimate $\boldsymbol{\phi}$ and joint angles from joint encoders $\boldsymbol{\Theta}_m$. Values $\boldsymbol{X}^{l_L,I}_m$ and $\boldsymbol{X}^{r_L,I}_m$ are the $m^{\text{th}}$ positions of the estimated left arm end-effector projected to left eye and right eye image coordinates, respectively, and $\boldsymbol{u}^{l_L}_m$ and $\boldsymbol{u}^{r_L}_m$ are the $m^{\text{th}}$ 2D position (pixel coordinates) of the left arm end-effector observed in the left and right eye/camera, respectively (variables $\boldsymbol{X}^{l_R,I}_m$, $\boldsymbol{X}^{r_R,I}_m$, $\boldsymbol{u}^{l_R}_m$ and $\boldsymbol{u}^{r_R}_m$ correspond to the right arm). Since the cost function contains both 3D and reprojection errors, the distances in space were multiplied by a coefficient $\mu$ determined from the intrinsic parameters of cameras and distance $d$ of the end-effector from the eye: $\mu = 320px/(d * (\pi/3))$.

### *D. Non-linear least squares optimization*

The objective functions (Eqs. [1]- [5]) defined for the optimization problem described in Section III-B are of the least-squares form and therefore can be minimized by Levenberg-Marquardt algorithm for non-linear least squares optimization (we used MATLAB implementation of the algorithm, same as in [17]). This iterative local algorithm performs minimization of a non-linear objective function by linearizing it at the current estimate every iteration. It interpolates between the Gauss-Newton and gradient descent method, combining advantages of both.

### *E. Error metrics*

For comparing the results achieved for individual settings, we make use of the following error metrics:

*1) Cartesian error between poses (position):* Cartesian position error $E_c$ between two generic poses, A and B, where $\boldsymbol{P_A} = [x_A, y_A, z_A]$ and $\boldsymbol{P_B} = [x_B, y_B, z_B]$ are 3D Cartesian positions of the end-effector, is defined as:

$$E_c = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2 + (z_A - z_B)^2}. \tag{6}$$

We evaluate the Cartesian error over the set of $N$ testing poses, which are selected as described in the section IV-B.

*2) Quality of estimated parameters:* For each estimated parameter $\phi_i$ we compute the mean difference ($e_i$) of the estimated parameter $\phi^e_i$ from the target parameter value $\phi^t_i$ (averaged over $R$ repetitions of the experiment):

$$e_i = \frac{\sum_{r=1}^{R} |\phi^e_{i,r} - \phi^t_i|}{R}, \tag{7}$$

as well as standard deviation of the parameter.

## IV. Data acquisition and description

### *A. Pose set generation*

With the goal of comparing different calibration methods on a humanoid robot, we chose a dataset where the two arms of the robot are in contact—thereby physically closing the kinematic chain through self-touch. At the same time, the robot gazes at the contact point (self-observation). The points were chosen from a cubic volume in front of the robot. For each target, using the Cartesian solver and controller [20], the iCub moves the left hand with end-effector in the palm to the specified point. Then it moves the right hand, with end-effector in the tip of the index finger, to the same point, with the additional constraint that the finger can be at most 50° away from the perpendicular direction of the palm. 5055 points and corresponding joint configurations were thus generated, with a difference on left and right effector position in every configuration of maximum $0.01mm$—see Fig. 2, right. The gaze controller [21] was used to command the neck and eyes of the robot to gaze at the same target (code and video can be accessed at [22]). The full dataset thus consists of 5055 data vectors $\boldsymbol{X}_i = [\boldsymbol{X}_i^{target}, \boldsymbol{X}_i^{RA}, \boldsymbol{X}_i^{LA}, \boldsymbol{\Theta}_i]$ composed of target point coordinates ($\boldsymbol{X}_i^{target} \in \mathbb{R}^3$), corresponding right arm and left arm end-effector positions ($\boldsymbol{X}^{RA} \in \mathbb{R}^3$, $\boldsymbol{X}^{LA} \in \mathbb{R}^3$), and joint angles $\boldsymbol{\Theta}_i$ for every joint of the torso, arms, neck, and eyes ($\boldsymbol{\Theta}_i \in \mathbb{R}^{20}$). Note that the solvers work with a given tolerance and hence $X_i^{target} \neq X_i^{RA} \neq X_i^{LA}$.

This way of dataset generation draws on previous work [2] and is hence feasible on the real robot provided sufficient quality of the initial model. Li et al. [14] provide an alternative control method: "tactile servoing". The robot could be also manipulated into the desired configurations while in gravity compensation mode.

### B. Training and testing dataset

We had 5055 configurations with $|\boldsymbol{X}_i^{RA} - \boldsymbol{X}_i^{LA}| < 0.01$ $mm$. The $0.01$ $mm$ error will at the same time constitute the lower bound on the maximum achievable calibration accuracy using the closure of the kinematic chain through self-touch. For the case of loop closure through the cameras, we employ the neck and eye joint values obtained from the solver in the simulator but reproject the end-effector positions directly and accurately into the cameras simulated in Matlab. The 5055 data points were further divided into training and testing datasets in the following way: $N$ out of 4755 poses are used as a training set on which the optimization process is performed (with a subset of 10, 20, 50, or 1000 poses chosen at random in different experiments) and 300 poses are used for testing purposes. Fig. 2, left, shows the distribution of joint values for individual joints in the dataset—this may impact the identifiability of individual parameters.

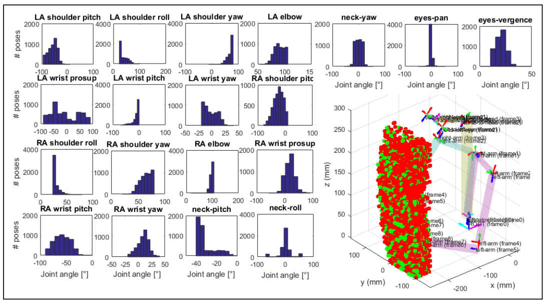


Fig. 2. Dataset visualization – 5055 configurations. (left) Distribution of joint values. (right) End-effector positions. Red – left arm; Green – right arm.

### C. Measurement error

Measurement noise with a Gaussian distribution was added motivated by the sensory accuracy in the real robot. Since distance between individual taxels on the real iCub sensitive skin is around 5 mm, we decided to use Gaussian noise with zero mean and $\sigma^2 = 5$ for touch as a baseline. For cameras, we introduce a 5px error (Gaussian noise with zero mean and $\sigma^2 = 5$ px), inspired by the setup in [3] where the iCub is detecting its fingertip in the camera frame. These errors are used in all experiments in the Simulation results section if not stated otherwise. In Section V-C we evaluate how changing the size of these measurement errors affects the resulting accuracy of end-effector position detection for individual chains.

### D. Perturbation of the initial parameters estimate

To evaluate the dependence of the optimization performance on the quality of the initial estimates of the parameters, we perturbed all estimated parameters by a *perturbation factor* $p = \{2, 5, 10, 20\}$. We perturbed all initial offset values $o_i$ as follows:

$$o_i^{new} = 1/100 * p * uniform[-1; 1] + o_i \; [rad], \tag{8}$$

It is reasonable to expect that the remaining DH parameters ($\alpha$, $a$, and $d$) will be in general more accurate as they can be extracted from CAD models and there is no moving part and no encoder involved. Therefore, their perturbation was chosen as follows:

$$\begin{aligned} \alpha : \alpha_i^{new} &= 1/1000 * p * uniform[-1; 1] + \alpha_i \; [rad], \\ a, d : \Phi_i^{new} &= 0.1 * p * uniform[-1; 1] + \Phi_i \; [mm]. \end{aligned} \tag{9}$$

## V. SIMULATION RESULTS

In this section we show the calibration results. We evaluated our approach using both error of the end-effector position—the cost function optimized (or distance in camera frame for projections into eyes)—as well as error in individual parameters (vs. their correct values). We compared kinematic chains used for calibration, number of free parameters which were estimated by the optimization process, different perturbation factor on individual parameters, number of training poses (data points), as well as measurement noise levels. Performance is always is evaluated on the testing dataset.

### A. Results for different chain combinations and number of training poses

Fig. 3 (top) shows the performance in terms of end-effector position estimation when DH parameters of the left arm (LA) chain are calibrated, utilizing different kinematic chain combinations: The "self-observation" from a single camera (LALEye) and "self-touch" only (LARA) are outperformed by "stereo self-observation" (LALREye) and all the chains together provide the best results (LARALREye). Clearly, more training poses (50 vs. 20) improve calibration results; 1000 poses should be sufficient to reach an optimal value and serve as a lower bound on the error. The effect of initial parameter perturbation factor is also shown; for all perturbation levels, the performance is stable (low error variance).

In Fig. 3 (bottom) only the largest "multi-chain" LARALREye is employed for training but the chains whose parameters are subject to calibration are varied. The error of end-effector position estimation is increasing with higher number of parameters estimated; however, even if parameters of all chains (86 DH parameters) are perturbed and subject to calibration

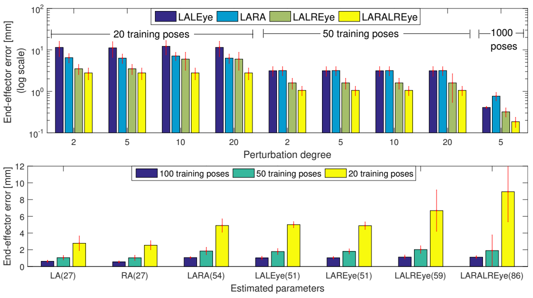


Fig. 3. End-effector position error after optimization—averaged over 10 repetitions. (Top) Left Arm chain calibration (full DH) using different chain combinations, different initial perturbation factors (2, 5, 10, 20) and training on 20 (left), 50 (middle), and 1000 poses (right – pert. factor 5 only). (Bottom) Performance of different parameter sets subject to calibration – LARALREye chain was used for calibration of parameters. Free parameters (being calibrated) in a given chain are denoted. E.g., LALEye denotes that all 51 DH parameters of left arm and left eye (including head) are calibrated, and the rest of the DH parameters (e.g. right arm) is considered to be known.

simultaneously, end-effector error of around 2 (1) $mm$ can be achieved with 50 (100) poses.

To investigate the distribution of errors for individual chains, we examined error residuals for every testing pose. For a higher number of training poses, error residuals have a zero mean and Gaussian distribution. For lower number of poses (especially for higher perturbation), the residuals are bigger and skewed and the resulting calibration also strongly depends on initialization. In Fig. 4, the end-effector error residuals for perturbation factor $p = 10$ are shown for their $x$ and $z$ coordinates (other 2D projections were qualitatively similar)—for different chains and different number of training poses.

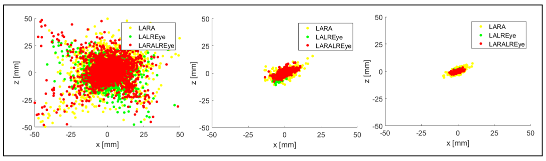


Fig. 4. Error residuals – Left Arm (LA) chain calibration using LARA, LALREye and LARALREye chains. Results visualized on 300 testing poses for each of 10 repetitions of the optimization, with random parameter initialization (3000 points in total per chain shown). (Left) 10 training poses; (Middle) 20 training poses; (Right) 50 training poses. Perturbation factor 10 and measurement errors 5 mm for skin and 5 px for cameras were considered.

## B. Observability analysis of individual chains

We conducted an observability analysis using Singular Value Decomposition (SVD) of the identification Jacobian matrix $J = [J_1, ..., J_n]$, where $n$ is the number of configurations in the training pose set and $J_n(i,j) = \left[\frac{\partial (X_i^r - X_i^e)}{\partial \phi_j}\right]$, $\phi_j$ is the parameter $j$ to be estimated, $(X_i^r - X_i^e)$ denotes the error between the real/observed ($X^r$) and estimated ($X^e$) value of the $i^{\text{th}}$ coordinate in the given chain.[1] The Jacobian matrix represents the sensitivity of end-effector positions or their camera reprojections to the change of individual parameters. Using SVD, we can obtain a vector of singular numbers $\sigma_i$. Comparison of the obtained singular numbers for individual chains for the task of estimating all DH parameters of the left arm (using same training pose set) can be seen in Fig. 5. We also evaluated observability indices $O_1$ [23] and $O_4$ [24] (performance of observability indices for industrial robot calibration was evaluated by Joubair [25]). $O_1$ index is defined as: $O_1 = \frac{(\sigma_1 \sigma_2 ... \sigma_m)^{1/m}}{\sqrt{(n)}}$, where $m$ is the number of independent parameters to be identified, $\sigma_i$ is the $i^{\text{th}}$ singular number, and $n$ is the number of calibration configurations. Index $O_4$ is defined as: $\frac{\sigma_m^2}{\sigma_1}$. See Fig. 5 (bottom panels). Te chain LALEye for 10 poses has very low observability caused by not full rank Jacobian (we have 24 parameters to estimate but only 20 equations). The highest observability is achieved in all cases for the largest chain LARALREye, where the information from touch and both cameras was used.

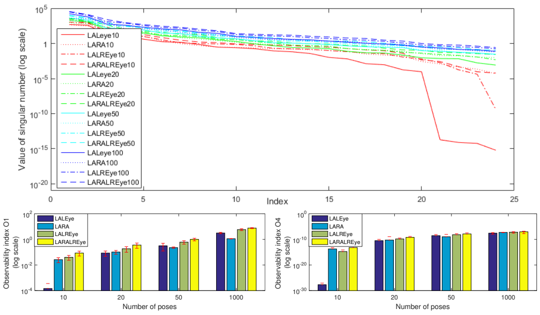


Fig. 5. Observability – Left Arm (LA) chain calibration (full DH) using different chain combinations. (Top) singular numbers of identification Jacobian for different chains used for calibration. Evaluation is performed over the same pose set for every chain. Red, green, turquoise, and blue color of the lines denote 10, 20, 50, and 1000 poses in the training set respectively. (Bottom left) Observability index $O_1$ [23]. (Bottom right) Observability index $O_4$ [24].

## C. Evaluation of error based on measurement noise

We evaluated the effect of measurement noise in individual sensors (touch, cameras) on the accuracy of end-effector position error on the testing data set—see Fig. 6. With same error in pixels on cameras and in $mm$ on "touch sensors" (first two columns – $2px/2mm$, $5px/5mm$), LALREye chain (both eyes, no touch) and LARALREye (both eyes and touch) have smallest final end-effector errors, for the "multi-chain" even smaller. When error on cameras increases ($5E2T$, $10E2T$, $10E5T$), the camera chains (LALEye, LALREye) are affected whereas the performance of the chain with touch (LARALREye) is not degraded. Conversely, more error on "touch" ($2E5T$, $2E10T$, $5E10T$) impacts the "touch only" chain (LARA), but the LARALREye remains robust.

## D. Quality of DH parameter estimates

To get further insight and take advantage of the simulation study where we have access to ground truth values of all

[1] E.g., for LALEye, $X$ corresponds to 2 errors: error on the coordinate $u$ and $v$ as a reprojection of the end-effector position into the cameras; for LARA chain, $X$ will correspond to 3 numbers: distance in x, y and z coordinate between right ($X^{r,R}$) and left arm ($X^{l,R}$) end-effector 3D positions.

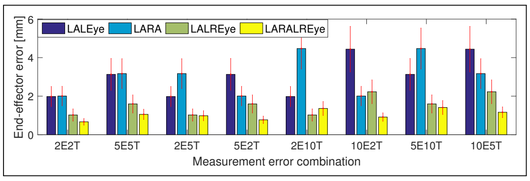


Fig. 6. End-effector position accuracy for different combinations of measurement noise on cameras and "touch sensor". Different chains employed to estimate DH parameters of the left arm (50 training poses, error evaluated over 300 testing poses, averaged over 10 repetitions). X-axis labels read as follows: first number – error on cameras ("Eyes") in pixels; second number – error on the touch sensor in $mm$ (i.e. 5E2T denotes that we introduced zero-mean Gaussian error with 5px and 2mm variance to cameras and touch respectively.

parameters, we also studied whether the optimization based on end-effector error also leads to correct estimates of all DH parameters—focusing on the left arm (LA) chain.

Fig. 7 shows the results for all estimated parameters when the LA-RA ("self-touch") chain was used for calibration, using different number of training poses. The errors on the length parameters (top panel) are on average distributed between approx. 1 and 10 $mm$. For the angular quantities, it is in the 0.1 to 1° range for the proximal joints.

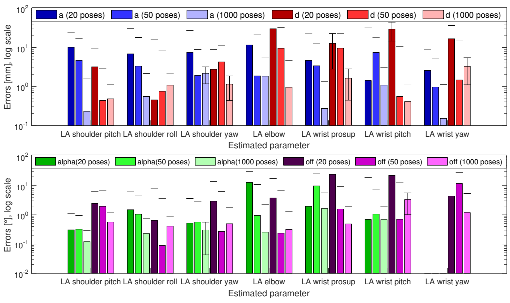


Fig. 7. Quality of DH parameter estimation for LA chain using LA-RA chain. Errors on individual parameters after optimization for different number of poses: (Top) $a$ and $d$ parameters; (Bottom) $\alpha$ and *offsets*. Averaged over 10 repetitions, perturbation factor 5, measurement noise $5px$ on cameras and $5mm$ on touch.

Finally, having showed above that the "self-touch and self-observation" (LARALREye) chain slightly outperforms the "stereo self-observation" only chain (LALREye) (Fig. 3 top, Fig. 6), also in observability (Fig. 5), here in Fig. 8 we can observe a similar trend in the estimated parameters of the LA chain against their ground truth values. The parameter estimates obtained from LARALREye are significantly better for $d$ for all joints except for wristPr and elbow and for $a$ for all shoulder joints. The other parameters estimates are comparable. The wrist joint calibration seems to be sensitive on the selection of training poses and will need further study.

## VI. Discussion and Conclusion

We quantitatively and systematically investigated the potential of automatic self-contained kinematic calibration (DH parameters including camera extrinsic parameters) of a humanoid robot employing different kinematic chains—in particular relying on self-observation and self-touch. The parameters varied were: (i) type and number of intersecting kinematic chains used for calibration, (ii) parameters and chains subject to optimization, (iii) amount of initial perturbation of kinematic parameters, (iv) number of poses/configurations used for optimization, (v) amount of measurement noise in end-effector positions / cameras. We also tracked the computation time and while the details differ depending on the settings (chain calibrated, number of poses, etc.), a typical optimization run would not take more than tens of seconds on an older laptop PC. Next to results w.r.t. the cost function itself (error on end-effector or camera reprojection) a number of additional analyses were performed including error residuals, errors on estimated parameters compared to ground truth, and observability analysis.

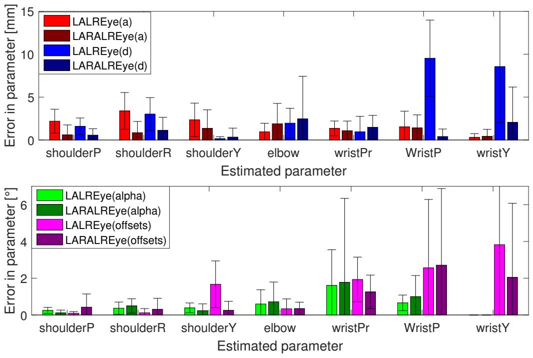


Fig. 8. Absolute error of estimated DH parameters of LA chain after optimization (50 training poses, perturbation factor 5, measurement noise 5 px on cameras and 5 mm on touch). (Top) $a$ and $d$ parameters. (Bottom) $\alpha$ and *offsets*.

While some results were expected (such as improvement when more configurations are added or poor performance when using self-observation from a single camera), the most notable findings are: (1) calibrating parameters of a single chain (e.g. one arm) by employing multiple kinematic chains ("self-observation" and "self-touch") is superior in terms of optimization results (Fig. 3 top) as well as observability (Fig. 5); (2) when using multi-chain calibration, fewer poses suffice to get similar performance compared to when e.g. only observation from a single camera is used (Fig. 3 top); (3) parameters of all chains (here 86 DH parameters) can be subject to calibration simultaneously and with 50 (100) poses, end-effector error of around 2 (1) mm can be achieved (Fig. 3 bottom); (4) adding noise to a sensory modality degrades performance of all calibrations employing the chains relying on this information (Fig. 6). The last point is interesting to discuss in relation to Birbach et al. [17] who put forth the hypothesis that calibrating multiple chains simultaneously is superior to pairwise sequential calibration. Our results support this provided that measurement noise is small. Instead, if a certain modality is noisy, it may be beneficial to preferentially employ chains that rely on more accurate measurements first and then calibrate a "noisy chain" in a second step.

We have only reported results from simulation, however, we claim that this was the right tool for this type of investigation. At the same time, our setup and choice of parameters was drawing on experiments performed in the real robot—self-touch [2] and self-observation [3], [4] in particular—which makes the results grounded in a real setting and should inform future experimentation on the iCub. The method to combine chains and analyze the results presented here can be transferred to other platforms as well.

There are several aspects that we want to further investigate in the future. First, we note that while we did control for the angle between the palm and the contralateral finger for self-touch in the dataset generation, we did not monitor whether the contact point would be also visible. Additional analyses revealed that the contact point would not be occluded and hence be visible by both cameras in 35% of the poses and by one of the cameras in 53%. We recomputed the observability with this subset of the dataset only and found no decrease. In the future, configurations with occlusions should be excluded from dataset generation. Second, we found that around 50 configurations (data points) suffice for reasonable calibration. Finding the optimal subset of not more than 10 configurations would be desirable, such that recalibration can be performed rapidly. Here, clever pose selection will be necessary to warrant adequate and stable performance. Third, the information from the two cameras can be used to reproject observed position of the end-effector in image coordinates of both eyes (pixel $(u, v)$) to 3D space ($X^{eye}$) (similar to [3], [26])—leading onto yet another formulation of the optimization problem. Fourth, our investigation can be extended considering also the contribution of inertial sensors—in the robot head [17] or distributed on the robot body [10], [8]. Fifth, the present method can be compared with filtering approaches [4], [16] or with methods that pose fewer assumptions on the initial model available (e.g., [27]). Finally, the self-touch scenario can be also turned around from using a tactile array to calibrate kinematics [2], [14] to calibrating the skin itself [28].

## ACKNOWLEDGMENT

We thank Alessandro Roncone for assistance with the models of the iCub robot in MATLAB and the source files leading to Fig. 1 left and Ugo Pattacini for discussions, tips, and assistance with the use of Cartesian solvers leading to the generation of self-touch configurations.

## REFERENCES

[1] J. Hollerbach, W. Khalil, and M. Gautier, "Model identification," in *Springer Handbook of Robotics*, 2nd ed., B. Siciliano and O. Khatib, Eds. Springer, 2016, pp. 113–138.

[2] A. Roncone, M. Hoffmann, U. Pattacini, and G. Metta, "Automatic kinematic chain calibration using artificial skin: self-touch in the iCub humanoid robot," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, 2014, pp. 2305–2312.

[3] S. R. Fanello, U. Pattacini, I. Gori, V. Tikhanoff, M. Randazzo, A. Roncone, F. Odone, and G. Metta, "3D stereo estimation and fully automated learning of eye-hand coordination in humanoid robots," in *2014 IEEE-RAS Int. Conf. on Humanoid Robots - HUMANOIDS '14*, 2014.

[4] P. Vicente, L. Jamone, and A. Bernardino, "Online body schema adaptation based on internal mental simulation and multisensory feedback," *Frontiers in Robotics and AI*, vol. 3, p. 7, 2016.

[5] 2019 (Project webpage). [Online]. Available: https://sites.google.com/site/karlastepanova/robot-self-calibration-icub

[6] M. Hersch, E. Sauser, and A. Billard, "Online learning of the body schema," *International Journal of Humanoid Robotics*, vol. 5, pp. 161–181, 2008.

[7] R. Martinez-Cantin, M. Lopes, and L. Montesano, "Body schema acquisition through active learning," in *Proc. Int. Conf. on Robotics and Automation (ICRA)*, 2010.

[8] P. Mittendorfer and G. Cheng, "Open-loop self-calibration of articulated robots with artificial skins," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 4539–4545.

[9] E. Dean-Leon, K. Ramirez-Amaro, F. Bergner, I. Dianov, and G. Cheng, "Integration of robotic technologies for rapidly deployable robots," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 4, pp. 1691–1700, 2018.

[10] N. Guedelha, N. Kuppuswamy, S. Traversaro, and F. Nori, "Self-calibration of joint offsets for humanoid robots using accelerometer measurements," in *Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on*. IEEE, 2016, pp. 1233–1238.

[11] K. Yamane, "Practical kinematic and dynamic calibration methods for force-controlled humanoid robots," in *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*. IEEE, 2011, pp. 269–275.

[12] C. Bartolozzi, L. Natale, F. Nori, and G. Metta, "Robots with a sense of touch," *Nature materials*, vol. 15, no. 9, pp. 921–925, 2016.

[13] R. S. Dahiya and M. Valle, *Robotic Tactile Sensing*. Springer, 2013.

[14] Q. Li, R. Haschke, and H. Ritter, "Towards body schema learning using training data acquired by continuous self-touch," in *Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on*. IEEE, 2015, pp. 1109–1114.

[15] R. Khusainov, A. Klimchik, and E. Magid, "Humanoid robot kinematic calibration using industrial manipulator," in *Mechanical, System and Control Engineering (ICMSC), 2017 International Conference on*. IEEE, 2017, pp. 184–189.

[16] R. Zenha, P. Vicente, L. Jamone, and A. Bernardino, "Incremental adaptation of a robot body schema based on touch events," in *Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, 2018.

[17] O. Birbach, U. Frese, and B. Bäuml, "Rapid calibration of a multi-sensorial humanoids upper body: An automatic and self-contained approach," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 420–436, 2015.

[18] 2017. [Online]. Available: http://wiki.icub.org/wiki/ICubForwardKinematics

[19] K. Nickels, "Hand-Eye calibration for Robonaut," NASA Summer Faculty Fellowship Program Final Report, Tech. Rep., 2003.

[20] U. Pattacini, F. Nori, L. Natale, G. Metta, and G. Sandini, "An experimental evaluation of a novel minimum-jerk Cartesian controller for humanoid robots," in *Proc. IEEE/RSJ Int. Conf. Int. Robots and Systems (IROS)*, 2010.

[21] A. Roncone, U. Pattacini, G. Metta, and L. Natale, "A Cartesian 6-DoF gaze controller for humanoid robots," in *Robotics: Science and Systems*, vol. 2016, 2016.

[22] 2017. [Online]. Available: https://github.com/matejhof/icub-selftouch-with-gaze-generator

[23] J.-H. Borm and C.-H. Menq, "Experimental study of observability of parameter errors in robot calibration," in *Robotics and Automation, 1989. Proceedings., 1989 IEEE International Conference on*. IEEE, 1989, pp. 587–592.

[24] A. Nahvi and J. M. Hollerbach, "The noise amplification index for optimal pose selection in robot calibration," in *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, vol. 1. IEEE, 1996, pp. 647–654.

[25] A. Joubair, A. Tahan, and I. A. Bonev, "Performances of observability indices for industrial robot calibration," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. IEEE, 2016, pp. 2477–2484.

[26] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 30, no. 2, pp. 328–341, 2008.

[27] P. Lanillos and G. Cheng, "Adaptive robot body learning and estimation through predictive coding," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, 2018.

[28] A. Albini, S. Denei, and G. Cannata, "Towards autonomous robotic skin spatial calibration: A framework based on vision and self-touch," in *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*. IEEE, 2017, pp. 153–159.

# Appendix I

# Multisensorial robot calibration framework and toolbox

Authors' version of:

Rozlivek, J., Rustler, L., Stepanova, K., and Hoffmann, M. (2021). Multisensorial robot calibration framework and toolbox. In *Humanoid Robots (Humanoids), IEEE-RAS International Conference on*, pages 459-466.

DOI: `https://doi.org/10.1109/HUMANOIDS47582.2021.9555803`
Youtube video: `https://youtu.be/ZZHztHF6eNs`

Author contributions: The contribution of M. Hoffmann was 20%.

# Multisensorial robot calibration framework and toolbox

Jakub Rozlivek[1,*], Lukas Rustler[1,*], Karla Stepanova[1,2], and Matej Hoffmann[1]

*Abstract*— **The accuracy of robot models critically impacts their performance. With the advent of collaborative, social, or soft robots, the stiffness of the materials and the precision of the manufactured parts drops and CAD models provide a less accurate basis for the models. On the other hand, the machines often come with a rich set of powerful yet inexpensive sensors, which opens up the possibility for self-contained calibration approaches that can be performed autonomously and repeatedly by the robot. In this work, we extend the theory dealing with robot kinematic calibration by incorporating new sensory modalities (e.g., cameras on the robot, whole-body tactile sensors), calibration types, and their combinations. We provide a unified formulation that makes it possible to combine traditional approaches (external laser tracker, constraints from contact with the external environment) with self-contained calibration available to humanoid robots (self-observation, self-contact) in a single framework and single cost function. Second, we present an open source toolbox for Matlab that provides this functionality, along with additional tools for preprocessing (e.g., dataset visualization) and evaluation (e.g., observability/identifiability). We illustrate some of the possibilities of this tool through calibration of two humanoid robots (iCub, Nao) and one industrial manipulator (dual-arm setup with Yaskawa-Motoman MA1400).**

## I. Introduction

Traditional industrial robots are composed of stiff materials, manufactured and assembled with high accuracy. With CAD drawings as a basis for their models, fine-tuning of parameters may be necessary for individual robot exemplars. Model identification [1] is a mature discipline. In this work, we focus on kinematic calibration. There are two main approaches [1]: (i) *open-loop* calibration, where the manipulator is not in mechanical (physical) contact with the environment and an external metrology system is used to measure robot pose components, and (ii) *closed-loop* calibration, where physical constraints on robot pose components replace external measurements. However, in their standard formulations, both approaches require a setup that is external to the robot itself. Also, typically, only one calibration method is applied at a time.

As robots are leaving controlled factory environments and starting to share workspaces with humans, the situation is changing. There is a fast growing market with collaborative, personal care, and social robots. These robots are cheaper, more lightweight, and less stiff. The precision of manufacturing and assembly drops and CAD models provide a less accurate basis for the robot models. On the other hand, the robots often come with a rich set of powerful yet inexpensive sensors like cameras, RGB-D cameras, inertial, tactile or force sensors. This opens up the possibility for calibration approaches that are more "self-contained", can be performed autonomously and repeatedly by the robot, and that simultaneously estimate the position of the sensors with respect to the robot.

Overall, there is a shift from highly accurate setups that span the whole lifetime of a robot to more adaptive settings: robots, including industrial robots, are re-deployed for different tasks, retrofitted with new sensors, and need to be repeatedly calibrated. The additional sensory equipment provides an opportunity for such calibration to be performed *in situ* and when needed. In this work, we provide a solution to these needs: a unified framework that brings together traditional robot kinematic calibration with the self-contained approaches exploiting sensors on the robot as well as possibilities available to humanoids or multi-arm setups like self-contact. We address also arbitrary combinations of these methods and provide an open-source toolbox [2] with this functionality and additional useful tools.

This article is structured as follows. After review of related work, Section III presents the theoretical formulation that unifies the different approaches to robot kinematic calibration, followed by the description of the toolbox (Section IV) with illustrative examples. We wrap up with Conclusion and future work.

An accompanying video illustrating some of the robot setups and functionality of the toolbox is available at https://youtu.be/ZZHztHF6eNs.

## II. Related work

Recently, automatic "self-contained" calibration approaches that can be performed autonomously by the robot have started to appear, typically relying on self-observation using the robot's own camera(s) (e.g., [3], [4], [5]) or additional sensors in the robot head [6]. Next to these open-loop approaches with camera(s) on the robot body, closed-loop approaches exploiting contact constraints can be also adapted to humanoid and similar platforms. With the advent of robotic skin technologies, self-touch can be exploited for calibration of robot kinematics [7], [8]. Another variant is exploiting sensitive fingertips to touch a known external surface [9].

This work was supported by the Czech Science Foundation (GA CR), project EXPRO (nr. 20-24186X). J.R. and L.R. were additionally supported by the Czech Technical University in Prague, grant No. SGS20/128/OHK3/2T/13.

[*] Both authors contributed equally.

[1] Jakub Rozlivek, Lukas Rustler, Karla Stepanova, and Matej Hoffmann are with Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic. matej.hoffmann@fel.cvut.cz.

[2] Karla Stepanova is also with Czech Institute of Informatics, Robotics, and Cybernetics, Czech Technical University in Prague, Czech Republic.

Birbach et al. [6] employed automated self-contained multisensorial calibration, although they only utilized sensors in the robot's head. Stepanova et al. [10] systematically studied on the simulated iCub humanoid robot how self-observation, self-contact, and their combination can be used for self-calibration. They found that employing multiple kinematic chains ("self-observation" and "self-touch") is superior in terms of optimization results as well as observability. In [11], a dual-arm industrial manipulator is calibrated using self-observation, self-contact, contact with external planar surfaces, and a laser tracker and the individual methods and their combinations are compared. Inertial sensing in the robot head [6] or distributed on the robot body [12], [13] can be also employed.

Traditionally, robot kinematics, tools / custom end effectors, or camera parameters are calibrated. With more sensors (cameras, RGB-D cameras, force/tactile sensors, or inertial measurement units) being added, their pose w.r.t. the robot model needs to be established as well. The robot is often retrofitted with such sensors and they thus do not feature in the CAD models. Artificial sensitive skins covering large areas of robot bodies constitute a specific case, as there may be hundreds or thousands of sensors, whose positions are not known. For the multimodal skin modules of Mittendorfer and Cheng, communication between modules and gravity projection in 3-axis accelerometers were employed to reconstruct every sensors' pose [14].

While toolboxes for camera calibration are abundant, there are few tools to support robot calibration. The Robot Toolbox [15] provides functions for the study and simulation of classical robotics focused on manipulators, but it does not address calibration. The python toolbox *pybotics* [16], [17] is specifically designed for calibration of robots' modified DH parameters by an external device. The C++ based "Robot Calibration" package for ROS [18] enables calibration of joint angle and robot frame offsets, and 3D Camera intrinsics and extrinsics by an external device (3D sensors) or by planar constraints. However, the unique capability of our toolbox is its versatility in terms of what is being calibrated (robot kinematics, camera parameters, whole-body skin) and with which method (external device, self-observation, contact constraints, self-contact—and their combinations).

## III. Multisensorial robot calibration

This section describes the theory necessary for multisensorial robot calibration. We build on top of our previous work [10], [11], but for the first time present a formulation that is agnostic to the robot used and that encompasses all the different calibration approaches.

### A. Robot representation

For calibration purposes, we expect a parametric model of the whole robot including any additional components like tools or artificial skin and sensory equipment (e.g., cameras, external measurement devices). Currently, we assume that all joints are revolute. The model is composed of $k \in N$ links, $N = \{1, .., n\}$, where every link is described by either:

1) Denavit-Hartenberg (DH) notation [19]:

$$\boldsymbol{\phi}_k = \{[a_k, d_k, \alpha_k, o_k]\}, \tag{1}$$

where $a_k$, $d_k$ and $\alpha_k$ are the first three parameters of the DH formulation of link $k$; $o_k$ is the offset that specifies the positioning of the encoders on the joints with respect to the DH representation.

2) Translation and rotation vector:

$$\boldsymbol{\phi}_k = \{[\boldsymbol{t_k}, \boldsymbol{r_k}]\}, \tag{2}$$

where link $k$ is described with a translation vector $\boldsymbol{t_k} = [t_{k,1}, t_{k,2}, t_{k,3}]^T$ and a rotation vector $\boldsymbol{r_k} = [r_{k,1}, r_{k,2}, r_{k,3}]^T$, with a unit vector for the axis of rotation $u_k = \frac{\boldsymbol{r_k}}{||\boldsymbol{r_k}||}$ and a rotation angle $\Theta_k = ||\boldsymbol{r_k}||$.

### B. Datasets

Every calibration approach requires different data: a self-touch dataset contains two contact points; self-observation dataset contains 2D projection points from cameras; external device dataset contains 3D coordinates of the observed point. To simplify data manipulation, we created a dataset format common for all the individual datasets. Each dataset point $\boldsymbol{D}_i$ from dataset $\boldsymbol{D}$ is defined as:

$$\boldsymbol{D}_i = [pn_i, \boldsymbol{\theta}_i, \boldsymbol{cp}_i, \boldsymbol{rp}_i, \boldsymbol{c}_i]$$

where $i \in \{1, .., M\}$ is an index identifying one particular dataset point, $M$ is the number of dataset points, $pn_i$ is pose number, which is used to unite dataset points with some shared property (e.g., photos from two cameras on the robot taken simultaneously), $\boldsymbol{\theta}_i$ is the current robot joint configuration (joint angles from joint encoders for the given robot configuration), $\boldsymbol{cp}_i$ are the contact (or observed) points (or point) defined by the specific robot frame and the translation in the frame, $\boldsymbol{rp}_i$ are the reference point coordinates in 3D or 2D (projection), $\boldsymbol{c}_i$ are the indexes of used cameras. A single robot configuration may be unfolded into multiple datapoints (with the same pose number).

The whole dataset $\boldsymbol{D}^{whole}$ is a set of individual datasets for different calibration approaches:

$$\boldsymbol{D}^{whole} = \{\boldsymbol{D}^{st}, \boldsymbol{D}^{pl}, \boldsymbol{D}^{so}, \boldsymbol{D}^{ed}\},$$

where $\boldsymbol{D}^{st}$, $\boldsymbol{D}^{pl}$, $\boldsymbol{D}^{so}$, and $\boldsymbol{D}^{ed}$ are datasets for self-touch (st), contact with planes (pl), self-observation (so), and external device measurements (ed), respectively.

### C. Multi-chain robot calibration

In multi-chain robot calibration, we estimate the parameter vector $\boldsymbol{\phi} = \{\boldsymbol{\phi}_k\}_{k \in K}$, where the parameters $\boldsymbol{\phi}_k$ defining each link $k$ might be expressed as in Eq. 1 or Eq. 2. $K \subseteq N$ is a subset of all robot links (e.g., links belonging to one robot arm). All parameters $\boldsymbol{\phi}_k$ of all links $k$ (then $K = N$) or their subset may be estimated. Calibration of only offsets $\mathbf{o}$ in joints is sometimes dubbed "daily calibration" [20].

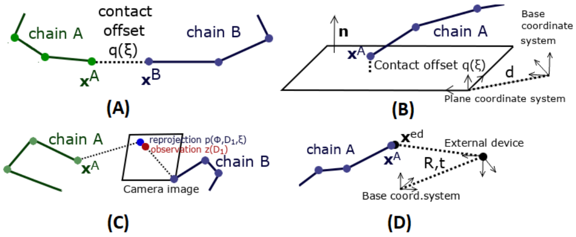


Fig. 1: Schematics of calibration using self-contact (A), contact with a plane (B), self-observation (C), and external device (D).

Estimation of the parameter vector $\boldsymbol{\phi}$ is done by optimizing a given objective function $\boldsymbol{f}(\boldsymbol{\phi}, \boldsymbol{D}, \boldsymbol{\zeta})$:

$$\boldsymbol{\phi}^* = \underset{\boldsymbol{\phi}}{\operatorname{argmin}}\, \boldsymbol{f}(\boldsymbol{\phi}, \boldsymbol{D}, \boldsymbol{\zeta}), \tag{3}$$

$$\boldsymbol{f}(\boldsymbol{\phi}, \boldsymbol{D}, \boldsymbol{\zeta}) = \|\boldsymbol{g}(\boldsymbol{\phi}, \boldsymbol{D}, \boldsymbol{\zeta})\|^2 = \sum_{i=1}^{M} g(\boldsymbol{\phi}, \boldsymbol{D}_i, \boldsymbol{\zeta})^2, \tag{4}$$

where $M$ is the number of robot configurations used for calibration (hereafter often referred to as "poses" for short), $\boldsymbol{\phi}$ is a given parameter estimate, dataset point $\boldsymbol{D}_i$ includes joint angles $\boldsymbol{\theta}_i$ from joint encoders for the given robot configuration, and constant vector $\boldsymbol{\zeta}$ defines all other necessary parameters such as camera calibration, fixed transformations, fixed kinematics parameters, or other properties of the robot.

In the following, we will often refer to individual kinematic chains. By a kinematic chain $A$ we mean a set of consecutive links $k$ ($k \in A$) starting with the robot base link. The position of the end of the chain in the robot workspace is determined by Cartesian coordinates $\boldsymbol{x}^A$ in the robot base frame. $\boldsymbol{X}^A$ notes a set of all $\boldsymbol{x}^A$ for the given dataset $D$. Let each of the links be described by parameters $\boldsymbol{\phi}_k$. Then we denote the set of parameters describing the whole kinematic chain as $\boldsymbol{\phi}^A$, $\boldsymbol{\phi}^A = \{\boldsymbol{\phi}_k\}_{k \in A}$.

During calibration, we typically calibrate one or multiple kinematic chains. This can be achieved through contact with the environment (e.g., touching a plane), by physically intersecting multiple chains (e.g., self-contact of two robot body parts) or by intersecting them virtually (e.g., using cameras – open-loop calibration).

Specific form of the function $g(\boldsymbol{\phi}, \boldsymbol{D}_i, \boldsymbol{\zeta})$ for individual considered chains and their combinations is discussed in the following subsections. In [10], [11], different combinations of intersecting chains were described in detail.

*1) Self-contact:* This corresponds to physical contact between individual parts of the robot itself. It can be contact between one arm and the skin on another arm [21], [8], [7], [10], contact between two end effectors on individual arms [11], or in general any contact between two robot parts (see Fig. 1, A).

In this case, the parameter vector $\boldsymbol{\phi}$ consists of the following parameters: $\boldsymbol{\phi} = \{\boldsymbol{\phi}^A, \boldsymbol{\phi}^B\}$, where $\boldsymbol{\phi}^A$ and $\boldsymbol{\phi}^B$ are kinematics parameters being calibrated corresponding to the first and second kinematic chain of the robot body, respectively. Let kinematic chains $A$ and $B$ end at the point $\boldsymbol{x}^A$ and $\boldsymbol{x}^B$ (e.g., tactile sensor, tip of finger or tool, etc.), respectively. Then, contact between these two chains (in the ideal case) means that $\boldsymbol{x}^A = \boldsymbol{x}^B$. The objective function to be optimized is:

$$\begin{aligned}\boldsymbol{g}^{st}(\boldsymbol{\phi}, \boldsymbol{D}^{st}, \boldsymbol{\zeta}) = &[c(\boldsymbol{\phi}, \boldsymbol{D}_1, \boldsymbol{\zeta}) - q(\boldsymbol{\zeta}), ..., \\ &c(\boldsymbol{\phi}, \boldsymbol{D}_M, \boldsymbol{\zeta}) - q(\boldsymbol{\zeta})]\end{aligned} \tag{5}$$

where the function $c(\boldsymbol{\phi}, \boldsymbol{D}_i, \boldsymbol{\zeta}) = ||\boldsymbol{X}_i^A(\boldsymbol{\phi}^A, \boldsymbol{D}_i, \boldsymbol{\zeta}) - \boldsymbol{X}_i^B(\boldsymbol{\phi}^B, \boldsymbol{D}_i, \boldsymbol{\zeta})||$ computes the distance of the points in contact of the two kinematic chains in the configuration given by the dataset point $\boldsymbol{D}_i$, where $\boldsymbol{D}_i \in \boldsymbol{D}^{st}$, where $\boldsymbol{D}^{st} \subset \boldsymbol{D}^{whole}$ is a set of dataset points with self-contact robot configurations. Parameter $q(\boldsymbol{\zeta})$ is the contact offset (e.g., thickness of the skin cover).

*2) Planar constraints – one arm chain in contact with a plane:* This corresponds to the scenario where the given robot body part ($A$) (e.g., robot arm) is getting into contact with a plane (see Fig. 1, B). In this type of optimization problem, we can distinguish formulations including single-plane or multiple-plane constraints [22], [23]. The classical formulations of the problem use either a general equation of the constraint plane or plane normals [24]. The general equation of a plane is $ax+by+cz+d = 0$, where $\mathbf{n} = (a, b, c)$ is a plane normal vector. The parameters of the plane can be known in advance (as in [24], [9] or [25] where calibration cube is used), or unknown [22].

When the plane parameters are unknown, the parameter vector $\boldsymbol{\phi}$ consists of the following parameters: $\boldsymbol{\phi} = \{\boldsymbol{\phi}^A, \mathbf{n}, d\}$, where $\boldsymbol{\phi}^A$ are the parameters of the robot kinematic chain in contact with the plane, $\mathbf{n}$ is a plane normal, and $d$ is the distance of the plane from the origin. We formulated the objective function as the distances between contacts and one or multiple fitted planes:

$$\begin{aligned}\boldsymbol{g}^p(\boldsymbol{\phi}^p, \boldsymbol{D}^p, \boldsymbol{\zeta}) = &[\mathbf{c}(\boldsymbol{\phi}^{p_1}, \boldsymbol{D}^{p_1}, \boldsymbol{\zeta}) - q(\boldsymbol{\zeta}), ..., \\ &\mathbf{c}(\boldsymbol{\phi}^{p_n}, \boldsymbol{D}^{p_n}, \boldsymbol{\zeta}) - q(\boldsymbol{\zeta})]\end{aligned} \tag{6}$$

where $\boldsymbol{D}^p \subset \boldsymbol{D}^{whole}$, $\boldsymbol{D}^p = \{\boldsymbol{D}^{p_1}, ..., \boldsymbol{D}^{p_n}\}$ is a set of datasets where contacts between the points in $\boldsymbol{X}^A$ and planes $p_1$,..., $p_n$ were performed. Parameter $q(\boldsymbol{\zeta})$ defines the contact offset (fixed distance between the contact point $\boldsymbol{x}^A$ on the robot and the plane). In direct contact, $q(\boldsymbol{\zeta}) = 0$; if the transformation directly to the contact point is not available, $q(\boldsymbol{\zeta}) \neq 0$. The set $\boldsymbol{\phi}^p = \{\boldsymbol{\phi}^{p_1}, ..., \boldsymbol{\phi}^{p_n}\}$ is a set of parameters for planes $p_1$, ..., $p_n$. The vector $\mathbf{c}(\boldsymbol{\phi}^{j,A}, \boldsymbol{D}^j, \boldsymbol{\zeta})$ is a vector of distances between individual end effector positions and the given plane $j$ for each datapoint $\boldsymbol{D}_i^j$ from the given dataset $\boldsymbol{D}^j$. The distance is computed using plane normals and corresponding plane coordinates as follows: $c(\boldsymbol{\phi}^{j,A}, \boldsymbol{D}_i^j, \boldsymbol{\zeta}) = ||\boldsymbol{n}^j \boldsymbol{p}_i^j(\boldsymbol{\phi}^{j,A}) + d||$. Point $\mathbf{p}_i^j$ is the position of the end effector computed by forward kinematics from dataset point $\boldsymbol{D}_i^j$; $\boldsymbol{\phi}^j = \{\boldsymbol{\phi}^{j,A}, \mathbf{n}^j, d^j\}$ is the estimated parameter vector. Finally, $\boldsymbol{\zeta}$ wraps up all other necessary parameters.

*3) Self-observation by cameras:* This corresponds to the scenario when we observe a point on the robot body with cameras which are part of the robot kinematic chain. The observed point might be an ArUCo marker [11], finger tip [3], light dot, taxel, etc. We can calibrate: (i) extrinsic parameters of the cameras (represented as standard robot links using DH (Eq. 1) or rototranslation representation (Eq. 2)) assuming the robot kinematic parameters to be known, (ii) the whole kinematic chain of the robot assuming camera extrinsic parameters are known, or (iii) the whole kinematic chain of the robot arm simultaneously with camera extrinsic parameters. Optimization is performed by minimizing the reprojection error between the observed point (e.g., ArUCo markers' positions in the camera) and the estimated position using the current kinematic model (see Fig. 1, C).

The parameter vector $\boldsymbol{\phi}$ consists of the following parameters: $\boldsymbol{\phi} = \{\boldsymbol{\phi}^A, \boldsymbol{\phi}^B\}$, where $\boldsymbol{\phi}^A$ are calibrated kinematics parameters of the kinematic chain ending in the observed point $\boldsymbol{x}^A$ on the given robot part and $\boldsymbol{\phi}^B$ are kinematic parameters of the kinematic chain ending in the robot camera. The objective function is formulated as the distance between projected points and their pixel coordinates in the images:

$$\begin{aligned} \boldsymbol{g}^{so}(\boldsymbol{\phi}, \boldsymbol{D}^{so}, \boldsymbol{\zeta}) =& [p(\boldsymbol{\phi}, \boldsymbol{D}_1, \boldsymbol{\zeta}) - z(\boldsymbol{D}_1), ..., \\ & p(\boldsymbol{\phi}, \boldsymbol{D}_{M'}, \boldsymbol{\zeta}) - z(\boldsymbol{D}_{M'})] \end{aligned} \tag{7}$$

where $p(\boldsymbol{\phi}, \boldsymbol{D}_i, \boldsymbol{\zeta})$ is the reprojection of the observed point $\boldsymbol{X}_i^A$ from dataset point $\boldsymbol{D}_i$ where $i \in \{1, ..., M'\}$, $M'$ is the number of individual observations over all robot configurations in the dataset $\boldsymbol{D}^{so}$. The $z(\boldsymbol{D}_i)$ is the actual observed point position in the camera image. This approach does not require information from both cameras/eyes and enables us to estimate only one side of the robot body (e.g., parameters of the left arm and left camera). For example, the estimated parameter vector $\boldsymbol{\phi}$ in the case of the kinematic chain connecting the left arm and the left camera consists of the following parameters: $\boldsymbol{\phi} = \{\boldsymbol{\phi}^A, \boldsymbol{\phi}^B\}$, where $\boldsymbol{\phi}^A$ and $\boldsymbol{\phi}^B$ are parameters corresponding to the robot left arm and to the left camera, respectively.

To obtain the reprojection of the point $\boldsymbol{X}_i^A$ into the camera coordinates, the observed point coordinates are transformed to the given camera frame using forward kinematics. Afterwards, a standard pinhole camera model extended with radial and tangential distortion coefficients [26] is applied to transform the 3D point in camera frame ($[x_c, y_c, z_c]^T$) into image coordinates $[u/w, v/w]^T$ (2D plane of the camera):

$$\begin{gathered} x'_c = x_c/z_c, \qquad y'_c = y_c/z_c, \qquad r = \sqrt{x'^2_c + y'^2_c} \\ \begin{bmatrix} x''_c \\ y''_c \end{bmatrix} = (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \begin{bmatrix} x'_c \\ y'_c \end{bmatrix} + \\ \begin{bmatrix} 2p_1 x'_c y'_c + p_2(r^2 + 2x'^2_c) \\ p_1(r^2 + 2y'^2_c) + 2p_2 x'_c y'_c \end{bmatrix} \\ \begin{bmatrix} u & v & w \end{bmatrix}^T = K \begin{bmatrix} x''_c & y''_c & 1 \end{bmatrix}^T \end{gathered} \tag{8}$$

where $K$ is camera matrix, $k_i$ are radial distortion coefficients, and $p_i$ are tangential distortion coefficients.

*4) Calibration via external devices – laser tracker/cameras:* Typically, a marker/retroreflector is attached to the robot and the robot is moved in its workspace while being observed by the external device. The distance between the 3D position of the marker acquired by the measurement device (laser tracker/cameras) and the 3D position computed from current robot arm kinematics parameters (plus current joint angle values using forward kinematics) is minimized (see Fig. 1, D).

The parameter vector $\boldsymbol{\phi}$ consists of the following parameters: $\boldsymbol{\phi} = \{\boldsymbol{\phi}^A, \boldsymbol{R}, \boldsymbol{t}\}$, where $\boldsymbol{\phi}^A$ are kinematics parameters corresponding to the calibrated body part with the attached marker/retroreflector (can be robot arm as well as leg or head), $\boldsymbol{R}$ and $\boldsymbol{t}$ are the rotation matrix and the translation vector defining the external device position w.r.t. the robot base frame. The objective function is formulated as the error of distances:

$$\boldsymbol{g}^{ed}(\boldsymbol{\phi}, \boldsymbol{D}^{ed}, \boldsymbol{\zeta}) = \left[p(\boldsymbol{\phi}, \boldsymbol{D}_1, \boldsymbol{\zeta}), ..., \; p(\boldsymbol{\phi}, \boldsymbol{D}_M, \boldsymbol{\zeta})\right], \tag{9}$$

where the function $p(\boldsymbol{\phi}, \boldsymbol{D}_i, \boldsymbol{\zeta}) = ||\boldsymbol{X}_i^{ed} - \boldsymbol{X}_i^A||$ computes the distance of the transformed point $\mathbf{X}_i^{ed}$ from external device (in its coordinate system) and the point $\mathbf{X}_i^A$ computed from forward kinematics and current estimate of robot kinematics parameters in the configuration given by the dataset point $\boldsymbol{D}_i$.

To calibrate the robot kinematics parameters, we minimize the distance between these 2 sets of 3D points using an iterative approach.

In each iteration of the optimization process we:

1) recompute the estimate of robot kinematic parameters
2) recompute rotation and translation matrix defining the external device position w.r.t. base frame. The relation between corresponding points in sets is generally:

$$\mathbf{X}_i^r = \boldsymbol{R}\mathbf{X}_i^{ed} + \boldsymbol{t} + N_i, \tag{10}$$

where $N_i$ is noise for the $i$-th datapoint. An algorithm introduced by Arun et al. [27] can be used for finding least-squares solution of $\boldsymbol{R}$ and $\boldsymbol{t}$. It is a non-iterative algorithm using singular value decomposition of a 3×3 matrix.

*5) Combining multiple chains:* In order to estimate all kinematic parameters of the robot, we can take advantage of combining some or all of the above-mentioned kinematic chains and correspondingly extend the parameter vector to be estimated.

The overall objective function can be generally defined as (depending on which datasets and criteria we want to use for calibration):

$$\begin{aligned} \boldsymbol{g}(\boldsymbol{\phi}, D, \boldsymbol{\zeta}) = \big[ & \boldsymbol{k}^{st} \odot \boldsymbol{g}^{st}(\boldsymbol{\phi}, \boldsymbol{D}^{st}, \boldsymbol{\zeta}), \boldsymbol{k}^{p} \odot \boldsymbol{g}^{p}(\boldsymbol{\phi}, \boldsymbol{D}^{p}, \boldsymbol{\zeta}), \\ & \boldsymbol{k}^{so} \odot \boldsymbol{g}^{so}(\boldsymbol{\phi}, \boldsymbol{D}^{so}, \boldsymbol{\zeta}), \boldsymbol{k}^{ed} \odot \boldsymbol{g}^{ed}(\boldsymbol{\phi}, \boldsymbol{D}^{ed}, \boldsymbol{\zeta}) \big], \end{aligned} \tag{11}$$

where $\boldsymbol{D}^{st}$, $\boldsymbol{D}^{p}$, $\boldsymbol{D}^{so}$, and $\boldsymbol{D}^{ed}$ are datasets for self-touch, planar constraints optimization, self-observation, and optimization via external devices, respectively. Symbol $\odot$ marks Hadamard product: $(\boldsymbol{k}^{st} \odot \boldsymbol{g}^{st})_i = k_i^{st} \cdot g_i^{st}$. Scale factor vectors $\boldsymbol{k}^j$ ($j \in \{st, p, so, ed\}$) allow combination of calibration approaches. It can be written as $k_i^j = c^j \cdot p \cdot \mu_i$, where $c^j$ reflects the reliability of the approach $j$ (e.g., $c^j = \sigma^{-\frac{1}{2}}$, where $\sigma$ is the uncertainty of the approach). The parameter $p$ is inversely proportional to the number of measurements for a given configuration to compensate for the number of added equations to calibration (e.g., multiple markers on the end effector detected by self-observation during one self-contact). The parameter $\mu_i$ transforms different units (e.g., pixels) to meters (is equal to one for all approaches except self-observation). The conversion is determined from intrinsic parameters of cameras as $\mu_i = d_i/f$, where $f$ is focal length and $d_i$ is the distance from camera to the $i$-th observed point.

### D. Non-linear least squares optimization

To solve the calibration task and find optimal parameters $(a^*, d^*, \alpha^*, o^*)$ or $(\boldsymbol{t^*}, \boldsymbol{r^*})$, an appropriate optimization technique has to be used. Robot kinematic calibration is generally cast as non-linear least squares parameter estimation [1] and solved using the Gauss-Newton or Levenberg-Marquardt algorithm. The latter—an improved version of the former—is used here, in line with state of the art in the literature. A reasonable initial estimate $\boldsymbol{\phi^i}$ of these parameters importantly contributes to the ability of the system to calibrate all the components and find a realistic robot model.

### E. Observability and identifiability of parameters

To evaluate the ability to calibrate a given robot model using the provided dataset, observability and identifiability of the parameters are used. According to [28], the observability index measures the quality of the dataset based on the identification Jacobian matrix $\mathbf{J}$, which represents the sensitivity of minimized values to the change of individual parameters. Borm and Menq [29] proposed a measure $O_1$; Driels and Pathre [30] proposed $O_2$; Nahvi and Hollerbach proposed measures $O_3$ [31] and $O_4$ [32]. All these measures can be computed from the singular value decomposition of $\mathbf{J}$. They are defined as:

$$O_1 = \frac{(\sigma_1 \sigma_2 \ldots \sigma_m)^{\frac{1}{m}}}{\sqrt{n}},\ O_2 = \frac{\sigma_{min}}{\sigma_{max}},\ O_3 = \sigma_{min},\ O_4 = \frac{\sigma_{min}^2}{\sigma_{max}},\ (12)$$

where $\sigma_j$ is the $j$-th singular number, $m$ is the number of independent parameters to be identified and $n$ is the number of calibration configurations.

The identification Jacobian matrix itself shows us the identifiability of individual optimized parameters: $\mathbf{J}(i,j) = \frac{\partial X_i}{\partial \phi_j}$, where $X_i$ is a distance (Eq. 5, Eq. 6, and Eq. 9) or a reprojection error (Eq. 7) and $\phi_j$ is the parameter to be estimated. If a matrix column related to a parameter consists only of zeros, the parameter is unidentifiable which leads to an unobservable problem (the minimal singular number is zero). According to [1], an unidentifiable parameter means that the experimental setup does not allow it to be identified, not that it is intrinsically unidentifiable.

### F. Perturbation of the initial parameters estimate

To evaluate the dependence of the optimization performance on the quality of initial parameter estimates, perturbation of individual parameters can be used. The perturbation factor $p$ is applied to individual parameters (DH parameters or rotation vectors):

$$\Phi_i^{new} = p \cdot uniform[-1;1] + \Phi_i\ [rad, m, -]. \qquad (13)$$

For example, in [11] and [10], we used $p = \{0.1, 0.3, 1\}$ for perturbation of offsets and $p = \{0.01, 0.03, 0.1\}$ for $\alpha, a, d$ as it is reasonable to expect that the DH parameters ($\alpha$, $a$, and $d$) will be in general more accurate as they can be extracted from CAD models and there is no moving part and no encoder involved.

### G. Evaluation

To evaluate the calibration quality, the dataset may be split into training and testing sets in a selected ratio (e.g., 70 to 30), each containing different robot poses. Optimization is based on the training data; testing data are used to evaluate the result.

The root-mean-square (RMS) error is used to compare results from multiple optimization runs. It is computed as:

$$RMS_c = \sqrt{\tfrac{1}{L}\textstyle\sum_{i=1}^{L}(k_i^c g^c(\boldsymbol{\phi}, \boldsymbol{D}_i^c, \boldsymbol{\zeta}))^2} = \sqrt{\tfrac{1}{L}\|\boldsymbol{k^c} \odot \boldsymbol{g^c}(\boldsymbol{\phi}, \boldsymbol{D}^c, \boldsymbol{\zeta})\|^2},\ (14)$$

where $L$ is the number of observations/measurements, $k_i$ is a scale factor for the given calibration approach $c$, and $g^c(\boldsymbol{\phi}, \boldsymbol{D}_i^c, \boldsymbol{\zeta})$ is the corresponding objective function (see Sec. III-C for specific form of individual objective functions).

To compare different executions when multiple chains are combined (see Sec. III-C.5), individual errors have to be summed:

$$RMS_{tot} = \sqrt{RMS_{so}^2 + RMS_{st}^2 + RMS_p^2 + RMS_{ed}^2}. \quad (15)$$

## IV. Software description

The theory described above was embedded in an open-source Multisensorial robot calibration toolbox [2]. Here we describe the main components of the toolbox. For details, see the README at [2].

### A. Software Architecture

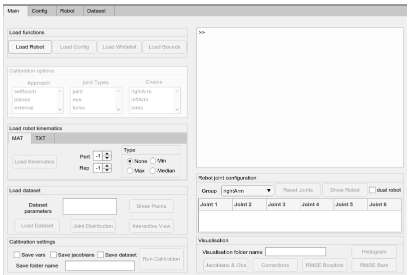


Fig. 2: GUI application screenshot.

The toolbox is a Matlab Add-On which provides functions and classes for robot calibration and a GUI application (see Fig. 2), which encapsulates the code into a working unit. The toolbox can be installed from Github [2] and used under GNU GPL license. Matlab (version R2016b and later) and the corresponding version of Matlab Optimization Toolbox is required. The pipeline and data flow are illustrated in Fig. 3. Fig. 4 provides a schematic overview.

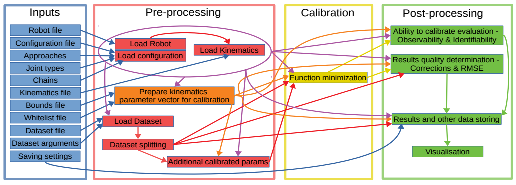


Fig. 3: Toolbox pipeline and data flow.

## B. Inputs

Calibration problems are defined by Robot description (*Robot*), Configuration settings (*Configuration*) and Dataset used for calibration (*Dataset*). Robot description might also optionally include *Whitelist* (which parameters should be calibrated) and *Bounds* (restrictions on the parameter values). The individual files are in a Matlab function format and return `struct` objects (more details below). Templates and examples of these input files are provided with the toolbox.

To simplify the definition of robot kinematic structure and the selection of parameters to calibrate, the possibility of grouping links is useful. For that reason, we prepared two data types. A data type $Group$ determines to which body part a link belongs (e.g., head), and $Type$ represents user-defined grouping across body parts, e.g., robot links and different skin parts (taxel, patch, triangle). The toolbox contains predefined values for both data types, but users can freely modify them or add new ones.

*1) Robot object:* *Robot* is considered as an object with a tree-like kinematic structure. The object is constructed based on three parameters defined in the Robot file: *robot name*, *links structure*, and *robot structure*. The *links structure* is an array of all robot links with their names, groups, link types, parents, and row indexes in the group-specific kinematic parameters matrices.

The *robot structure* consists of: 1) *Kinematics parameters*: Description of the robot kinematics (in metres and radians; more details in Sec. III-A); 2) *Eyes (cameras)*: Description of robot eyes (mounted cameras) by their camera matrix, radial and tangential distortion coefficients (see Eq. 8); 3) *Whitelist*: Boolean matrices determining which kinematics parameters should be calibrated. 4) *Bounds*: Bounds for parameters in case of calibration with bounded fitting parameters.

The robot can be visualized with cylinders for every robot link (derived from [33]).

*2) Configuration:* Calibration properties and settings are determined by the *Configuration* containing: 1) Optimization solver settings (Matlab optimization toolbox – e.g., algorithm, maximum number of iterations, termination tolerance); 2) Calibrated chain: which body parts ($Group$) will be calibrated; 3) Calibration approach (more details in Sec. III-C); 4) Calibrated link type: which $Type$ will be calibrated; 5) Perturbations: perturbation ranges for each type of parameter ($[a, d, \alpha, o]$ or $[t_1, t_2, t_3, r_1, r_2, r_3]$) as defined in Sec. III-F; 6) Calibration settings (e.g., number of repetitions, units of length, train/test dataset part split ratio).

*3) Dataset:* The toolbox expects calibration datasets in the format described in Sec. III-B. All datasets contain *pose-number* and *robot joint configuration*. Depending on the calibration type, the dataset also includes other information (e.g., self-contact calibration contains also *contact frame*, *second contact frame*, *translation in contact frames* (points), *reference coordinates of the contact point* (optional), and *reference distance from the contact point* (optional)).

Multiple datasets can be merged into one dataset structure containing datasets for different calibration types (self-observation, self-contact, etc.) in the corresponding structure field. As the dataset file is a Matlab function, calibration settings, calibrated chains, and additional function arguments can be used for faster dataset structure preparation.

## C. Preprocessing

Before calibration, input files have to be preprocessed – see Fig. 3. First, individual input files are loaded (robot parameters, configuration, and kinematics; see Sec. IV-B). Second, the robot object is constructed. Third, based on the calibration properties and settings (information of selected calibration approaches, joint types and chains to be calibrated, whitelist of calibrated parameters, etc.), calibration data are prepared. This is complemented by optional application of perturbation and bound parameters. If perturbation is chosen, only the calibrated parameters are perturbed (Sec. III-F for details). Then the original and perturbed parameters are extended with their upper and lower bounds. If a perturbed parameter is out of bounds, then it is set to the bound value. Finally, when the robot and calibration properties are prepared, the dataset is loaded. As mentioned in Sec. III-G, the datasets are divided into a training part and a testing part. The split is done based on the pose number, which means that all data points of the same pose (with the same pose number) are in the same part.

There are several dataset visualizations available: 1) histogram of distribution of robot joint angle values; 2) interactive visualization of individual joint configurations from the dataset; 3) robot in its default joint configuration and the data points coordinates computed from the robot kinematic parameters. See the corresponding panel in Fig. 4.

Plane parameters (planar calibration) or the transformation matrix from the external device to the robot base frame (external calibration) may be unknown and can be added to calibration (more details in Sec. III-C). In that case, the initial parameters are calculated for the training dataset and added to the vector of calibrated parameters before calibration.

Results of previous calibrations can be loaded from .mat or text file as the starting parameters for the current calibration.

## D. Calibration

Calibration is provided by the nonlinear least-squares solver *lsqnonlin* from Matlab Optimization Toolbox. The minimized function returns a vector of function values computed for a chosen subset of functions from Sec. III-C. Calibration runs for a specific number of repetitions with differently divided datasets into training and testing part. All repetitions are done for parameters without perturbation, and eventually, for perturbed parameters based on the perturbation levels prepared and chosen in *Configuration*.

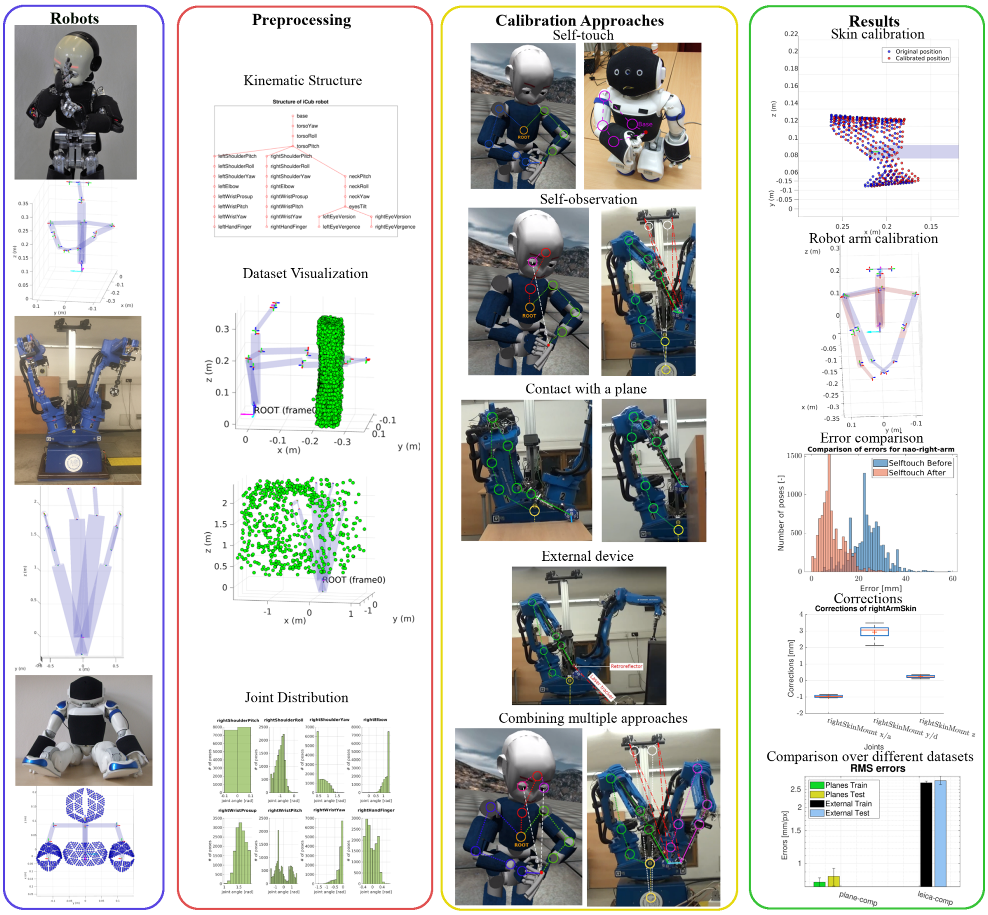


Fig. 4: Schematic overview and examples of the toolbox pipeline and functionality.

### E. Postprocessing and results

The ability to calibrate the selected kinematic parameters is evaluated by identifiability and observability of calibrated parameters (more details in Sec. III-E). The Jacobian matrices are returned directly from the optimization solver, and the observability indexes are computed for each iteration.

The results of calibration are evaluated by: 1) computing corrections of the calibrated parameters over multiple repetitions; and 2) RMSE errors for training and testing data before and after calibration (more details in Sec. III-G).

Most of the variables and function arguments (e.g., calibrated kinematics results, corrections, RMSE) are saved in .mat files. Additionally, the calibrated kinematics results and calibration properties and settings are saved in text files. Large files like Jacobian matrices are saved only on request.

Prepared visualizations can show the saved results (e.g., Fig. 4). The boxplots (implementation by [34]) are used to visualize Jacobians, observability indexes (Eq. 12), corrections, and RMSE (Eq. 14). Moreover, the errors can be shown in a bar chart (RMSE), error-joint dependence plot, and histogram of errors before and after calibration.

### F. Illustrative examples

A schematic overview is in Fig. 4. To demonstrate how the toolbox works, we show calibration of three different robots (first column in Fig. 4): iCub humanoid robot (see [10] for details), Nao humanoid robot with whole-body artificial skin [35], and dual-arm setup with Yaskawa-Motoman MA1400 [11]. The repository [2] includes also other examples that can be used as templates. In the preprocessing step, one can visualize mainly dataset related plots. Examples of iCub tree-like structure, iCub and Motoman dataset points, and iCub joint distribution visualizations are shown in the second column of Fig. 4.

Different robots can be calibrated by different approaches and their combinations as shown in the third column. The last column presents results of successful calibration for some of the approaches mentioned above. The following cases are shown: spatial calibration of the artificial skin, robot kinematic calibration (iCub right arm), histogram of errors before and after calibration of skin on the Nao using self-touch and corrections of rightArm skin parameters, and RMS errors comparison of Motoman calibration by planes and by external device.

## V. Conclusion and future work

We presented a framework for robot kinematic calibration which extends state-of-the-art methods of open-loop and closed-loop calibration [1] by providing a unifying formulation that encompasses approaches typical for industrial robots (laser tracker, physical contact with environment) as well as "self-contained" calibration approaches available to humanoids (self-observation, self-contact). We present a formulation that covers all these types and allows for their combinations. Second, the framework is embedded in an open-source toolbox for Matlab and we demonstrate its functionality on several robots and calibration approaches. Unique to our framework and software, we also support calibration of/using whole-body tactile sensors.

In the future, we plan to extend our framework and toolbox to include also the use of inertial sensors as shown in [6], [12], [13]. Next to revolute, prismatic joints could be also added. Planar constraints are only one example of closed-loop calibration approaches—other variants that constrain a different number of degrees of freedom are possible and can be added. From a practical point of view, it would be beneficial to enable importing robot kinematics in the Unified Robot Description Format (URDF) and reflect measurement uncertainty of each measured pose in scale factors.

## References

[1] J. Hollerbach, W. Khalil, and M. Gautier, "Model identification," in *Springer Handbook of Robotics*, 2nd ed., B. Siciliano and O. Khatib, Eds. Springer, 2016, pp. 113–138.

[2] J. Rozlivek, L. Rustler, K. Stepanova, and M. Hoffmann, 2020. [Online]. Available: https://github.com/ctu-vras/multirobot-calibration

[3] S. R. Fanello, U. Pattacini, I. Gori, V. Tikhanoff, M. Randazzo, A. Roncone, F. Odone, and G. Metta, "3D stereo estimation and fully automated learning of eye-hand coordination in humanoid robots," in *Humanoid Robots (Humanoids), IEEE-RAS Int. Conference on*, 2014.

[4] R. Martinez-Cantin, M. Lopes, and L. Montesano, "Body schema acquisition through active learning," in *Robotics and Automation (ICRA), IEEE International Conference on*, 2010.

[5] P. Vicente, L. Jamone, and A. Bernardino, "Online body schema adaptation based on internal mental simulation and multisensory feedback," *Frontiers in Robotics and AI*, vol. 3, p. 7, 2016.

[6] O. Birbach, U. Frese, and B. Bäuml, "Rapid calibration of a multi-sensorial humanoid's upper body: An automatic and self-contained approach," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 420–436, 2015.

[7] A. Roncone, M. Hoffmann, U. Pattacini, and G. Metta, "Automatic kinematic chain calibration using artificial skin: self-touch in the iCub humanoid robot," in *Robotics and Automation (ICRA), IEEE International Conference on*, 2014, pp. 2305–2312.

[8] Q. Li, R. Haschke, and H. Ritter, "Towards body schema learning using training data acquired by continuous self-touch," in *Humanoid Robots (Humanoids), IEEE-RAS Int. Conf. on*. IEEE, 2015, pp. 1109–1114.

[9] R. Zenha, P. Vicente, L. Jamone, and A. Bernardino, "Incremental adaptation of a robot body schema based on touch events," in *Joint IEEE Int. Conf. on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*, 2018.

[10] K. Stepanova, T. Pajdla, and M. Hoffmann, "Robot self-calibration using multiple kinematic chains – a simulation study on the iCub humanoid robot," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1900–1907, 2019.

[11] K. Stepanova, J. Rozlivek, F. Puciow, P. Krsek, T. Pajdla, and M. Hoffmann, "Automatic self-contained calibration of an industrial dual-arm robot with cameras using self-contact, planar constraints, and self-observation," *Robotics and Computer-Integrated Manufacturing*, 2021. [Online]. Available: http://arxiv.org/abs/2012.07548

[12] N. Guedelha, N. Kuppuswamy, S. Traversaro, and F. Nori, "Self-calibration of joint offsets for humanoid robots using accelerometer measurements," in *Humanoid Robots (Humanoids), IEEE-RAS International Conference on*. IEEE, 2016, pp. 1233–1238.

[13] P. Mittendorfer and G. Cheng, "Open-loop self-calibration of articulated robots with artificial skins," in *Robotics and Automation (ICRA), IEEE International Conference on*. IEEE, 2012, pp. 4539–4545.

[14] P. Mittendorfer and G. Cheng, "3D surface reconstruction for robotic body parts with artificial skins," in *Intelligent Robots and Systems (IROS), IEEE/RSJ Int. Conference on*, Oct 2012, pp. 4505–4510.

[15] P. I. Corke, *Robotics, Vision & Control: Fundamental Algorithms in MATLAB*, 2nd ed. Springer, 2017, iSBN 978-3-319-54413-7.

[16] N. Nadeau, "Pybotics: Python toolbox for robotics." [Online]. Available: https://github.com/nnadeau/pybotics

[17] ——, "Pybotics: Python toolbox for robotics," *Journal of Open Source Software*, vol. 4, no. 41, p. 1738, Sept. 2019.

[18] M. Ferguson, "Robot calibration." [Online]. Available: https://github.com/mikeferguson/robot_calibration

[19] J. Denavit and R. S. Hartenberg, "A kinematic notation for lower pair mechanisms based on matrices," *Journal of Applied Mechanics*, vol. 77, no. 2, pp. 215–221, 1955.

[20] K. Nickels, "Hand-Eye calibration for Robonaut," NASA Summer Faculty Fellowship Program Final Report, Tech. Rep., 2003.

[21] A. Albini, S. Denei, and G. Cannata, "Towards autonomous robotic skin spatial calibration: A framework based on vision and self-touch," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*. IEEE, 2017, pp. 153–159.

[22] H. Zhuang, S. H. Motaghedi, and Z. S. Roth, "Robot calibration with planar constraints," in *Robotics and Automation (ICRA), IEEE International Conference on*, vol. 1. IEEE, 1999, pp. 805–810.

[23] A. Joubair and I. A. Bonev, "Non-kinematic calibration of a six-axis serial robot using planar constraints," *Precision Engineering*, vol. 40, pp. 325–333, 2015.

[24] M. Ikits and J. M. Hollerbach, "Kinematic calibration using a plane constraint," in *Robotics and Automation (ICRA), IEEE International Conference on*, vol. 4. IEEE, 1997, pp. 3191–3196.

[25] A. Joubair and I. A. Bonev, "Kinematic calibration of a six-axis serial robot using distance and sphere constraints," *The Int. Journal of Adv. Manufacturing Technology*, vol. 77, no. 1-4, pp. 515–523, 2015.

[26] J. Heikkila and O. Silven, "A four-step camera calibration procedure with implicit image correction," in *Proc. of IEEE Comp. Soc. Conf. on Computer Vision and Pattern Recognition*, 1997, pp. 1106–1112.

[27] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-d point sets," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, no. 5, pp. 698–700, 1987.

[28] Y. Sun and J. M. Hollerbach, "Observability index selection for robot calibration," in *Robotics and Automation (ICRA), IEEE International Conference on*. IEEE, 2008, pp. 831–836.

[29] J.-H. Borm and C.-H. Menq, "Experimental study of observability of parameter errors in robot calibration," in *Robotics and Automation (ICRA), IEEE International Conference on*. IEEE, 1989, pp. 587–592.

[30] M. R. Driels and U. S. Pathre, "Significance of observation strategy on the design of robot calibration experiments," *Journal of Robotic Systems*, vol. 7, no. 2, pp. 197–223, 1990.

[31] A. Nahvi, J. M. Hollerbach, and V. Hayward, "Calibration of a parallel robot using multiple kinematic closed loops," in *Robotics and Automation (ICRA), Int. Conference on*. IEEE, 1994, pp. 407–412.

[32] A. Nahvi and J. M. Hollerbach, "The noise amplification index for optimal pose selection in robot calibration," in *Robotics and Automation (ICRA), IEEE International Conference on*, vol. 1. IEEE, 1996, pp. 647–654.

[33] "Wiki for iCub and friends – forward kinematics," 2017. [Online]. Available: http://wiki.icub.org/wiki/ICubForwardKinematics

[34] J. C. Lansey, "Box and whiskers plot (without statistics toolbox)," 2020, retrieved December 4, 2020. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/42470-box-and-whiskers-plot-without-statistics-toolbox

[35] L. Rustler, B. Potocna, M. Polic, K. Stepanova, and M. Hoffmann, "Spatial calibration of whole-body artificial skin on a humanoid robot: comparing self-contact, 3D reconstruction, and CAD-based calibration," in *Humanoid Robots (Humanoids), IEEE-RAS International Conference on*, 2021.

# Appendix J

# Spatial calibration of whole-body artificial skin on a humanoid robot: comparing self-contact, 3D reconstruction, and CAD-based calibration

Authors' version of:

Rustler, L., Potocna, B., Polic, M., Stepanova, K., and Hoffmann, M. (2021). Spatial calibration of whole-body artificial skin on a humanoid robot: comparing self-contact, 3D reconstruction, and CAD-based calibration. In *Humanoid Robots (Humanoids), IEEE-RAS International Conference on*, pages 445-452.

DOI: `https://doi.org/10.1109/HUMANOIDS47582.2021.9555806`<br>
Youtube video: `https://youtu.be/CCa2OPDq-BY`

Author contributions: The contribution of M. Hoffmann was 20%.

# Spatial calibration of whole-body artificial skin on a humanoid robot: comparing self-contact, 3D reconstruction, and CAD-based calibration

Lukas Rustler[1], Bohumila Potocna[1], Michal Polic[1,2], Karla Stepanova[1,2], and Matej Hoffmann[1]

***Abstract*— Robots were largely missing the sense of touch for decades. As artificial sensitive skins covering large areas of robot bodies are starting to appear, to be useful to the machines, sensor positions on the robot body are needed. In this work, a Nao humanoid robot was retrofitted with pressure-sensitive skin on the head, torso, and arms. We experimentally compare the accuracy and effort associated with the following skin spatial calibration approaches and their combinations: (i) combining CAD models and skin layout in 2D, (ii) 3D reconstruction from images, (iii) using robot kinematics to calibrate skin by self-contact. To acquire 3D positions of taxels on individual skin parts, methods (i) and (ii) were similarly laborious but 3D reconstruction was more accurate. To align these 3D point clouds with the robot kinematics, two variants of self-contact were employed: skin-on-skin and utilization of a custom end effector (finger). In combination with the 3D reconstruction data, mean calibration errors below the radius of individual sensors were achieved (2 mm). Significant perturbation of more than 100 torso taxel positions could be corrected using self-contact calibration, reaching approx. 3 mm mean error. This work is not a proof of concept but deployment of the approaches at scale: the outcome is actual spatial calibration of all 970 taxels on the robot body. As the different calibration approaches are evaluated in isolation as well as in different combinations, this work provides a guideline applicable to spatial calibration of different sensor arrays.**

## I. Introduction

Tactile sensing is crucial to improve robot capabilities beyond state of the art performance. Artificial skins covering large areas of robot bodies with sensitivity to force/pressure make it possible to unambiguously localize multiple simultaneous contacts which can be exploited for robot control in safety-related ways (physical Human-Robot Interaction) or in social HRI. The importance of endowing robots with touch has been recognized for several decades and a large number of technologies have been developed (see e.g., the 2019 special issue of Proceedings of the IEEE [1]). The focus has largely been on tactile sensing for manipulation, as equipping robot only hands/fingers with tactile sensors requires relatively small patches of electronic skin. Whole-body artificial skins for robots have been an exception, with only a few successful technologies deployed on complete robots and over extended time periods: (i) capacitive skin coming from the ROBOSKIN project [2] was used for example on the iCub humanoid; (ii) PVDF films were embedded beneath the skin of CB$^2$ robot [3]; (iii) photo-reflectors covered by urethane foam [4] were later used on the baby humanoid Noby; (iv) the multimodal skin modules of Mittendorfer, Cheng, and colleagues [5] were deployed on a number of different robots. The potential of sensing over the whole body surface has also not been fully explored, but Cheng et al. provide an overview of applications of their electronic skin [6]. Recently, artificial skin solutions have found their way to the collaborative robot industry through Airskin (pressure-sensitive) and Bosch APAS (using proximity).

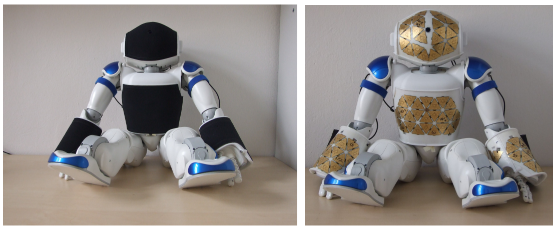


Fig. 1. Nao with artificial skin. (Left) With complete skin. (Right) Without top layer—skin patches with triangles and taxels exposed.

For large area tactile arrays to be useful, their spatial calibration—determining the 3D location of individual sensors with respect to known reference frames on the robot—is indispensable. Often, the sensor arrays are fitted onto existing robots and placed manually. With skin arrays containing hundreds or thousands of sensors, manual registration of their positions is not feasible. We review the approaches used in the past and then we provide an experimental evaluation of several methods on what we consider a representative example: a Nao humanoid robot has been retrofitted with additional 3D printed covers to host skin arrays on its head, torso, and hands. The capacitive tactile sensors, 970 in total, have then been manually placed onto these mounts and bent to conform to the 3D surface. We compare the accuracy and effort associated with the following calibration approaches and their combinations: (i) combining CAD models and skin layout in 2D, (ii) 3D reconstruction from images, (iii) using robot kinematics to calibrate skin using self-contact.

An accompanying video illustrating details of the methods and results is available at `https://youtu.be/CCa2OPDq-BY`.

This work was supported by the Czech Science Foundation (GA CR), project EXPRO (No. 20-24186X). L.R. was additionally supported by the Czech Technical University in Prague (No. SGS20/128/OHK3/2T/13). M.P. was supported by the EU H2020 project ARtwin (No. 856994). We thank Pavel Krsek, Tomas Pajdla, Alessandro Albini, Jakub Rozlivek, Tomas Svoboda, and Silvio Traversaro for inputs. Martin Hoffmann contributed CAD processing and the custom end effector. Hassan Saeed contributed scripts for importing and visualizing taxel position data. Maksym Shcherban assisted with CAD files and visualizations.

[1] Lukas Rustler, Bohumila Potocna, Michal Polic, Karla Stepanova, and Matej Hoffmann are with Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic. `matej.hoffmann@fel.cvut.cz`.

[2] Michal Polic and Karla Stepanova are also with Czech Institute of Informatics, Robotics, and Cybernetics, CTU Prague, Czech Republic.

## II. Related work

Robot kinematic calibration aims at identifying parameters such as link lengths and orientations of the robot structure and any added tools [7]. Different mechanical systems like measurement arms or contactless measurement systems like laser trackers are employed. For calibrating large-area tactile arrays on the robot body, the same methods partially apply, but the number of parameters grows enormously: every tactile element (*taxel* in what follows) forms its own link extending the robot kinematics. For example, mechanical measurement arms would be applicable to spatial calibration of large-area sensor arrays. However, with hundreds or thousands of sensors, such a procedure would be extremely laborious. Laser trackers and visual-based methods rely on special markers that are typically larger than the skin taxels. Identification of individual taxels in a camera image depends on the appearance of the sensors and whether they are exposed or covered by a protective layer.

Some artificial skins feature multiple sensing modalities. The multimodal skin modules of Mittendorfer and Cheng [5] host 3-axis accelerometers and communication capabilities that allow the modules to identify their neighbors and hence establish the skin topology. Combining these and knowledge about the modules' dimensions for skin covering a single body part, it is possible to reconstruct every sensors' pose using a touch-less method by sampling of gravity vectors in multiple robot poses [8]. In [9], this work is extended exploiting the LEDs also embedded in the modules: selectively turning on different modules' LEDs with different colors, salient visual features are formed. When observed by an external monocular camera, 6D pose of such "adaptive visual markers" can be estimated. However, our focus is on tactile arrays that have pressure-sensing capability only.

Cannata et al. [10] formulate skin spatial calibration as a maximum likelihood mapping problem in 6D space, where both the position and orientation of every taxel is recovered. In their theoretical account, they show how this could be achieved when the tactile sensors contact external objects with a known pose with respect to the robot.

It is desirable to have a calibration method that is self-contained and does not require any external mechanical or sensing equipment. For humanoid robots or dual-arm setups, self-contact or self-touch constitutes such a method. Assuming the taxel positions are known, self-contact can be employed for calibrating robot kinematics ([11], [12]; in combination with self-observation [13]). Conversely, if the kinematics is known and the skin array positions are not—which is more often the case—self-contact can be deployed to calibrate the skin, as shown by Albini et al. [14] and here.

Spatial calibration of pressure-sensitive skin of the kind used here (derived from [2]) has been performed in the past. Del Prete et al. [15] inferred the taxel poses by estimating contact points from interaction forces when the iCub arm was contacting an external point probe. However, this method requires force/torque sensing located proximally to the skin array being calibrated and a model of the robot kinematics and dynamics. The work most related to ours is Albini et al. [14] who employed self-contact. The main differences are: (i) touch was performed autonomously by the Baxter robot relying on the kinematic model and an RGB-D camera; (ii) contact was performed with a finger/stylus of radius 1 cm; (iii) every taxel's position was calibrated independently (122 taxels on the robot forearm were tested). In contrast, we perform skin spatial calibration on four body parts of the robot comprising 970 taxels in total and experimentally compare several different methods and their combinations (prior knowledge from CAD model and 2D skin layout, 3D reconstruction from images, and two variants of self-contact). We move the robot into the self-contact positions manually.

## III. Experimental setup

### A. Nao humanoid robot with pressure-sensitive skin

We used the Nao, Evolution (V5, Nao H25 V50). From kinematic parameters from the manufacturer, we developed a model of the robot's upper body using the Denavit-Hartenberg convention (Table 2.1 in [16] for details).

The robot used, shown in Fig. 1, is uniquely equipped with artificial skin—same kind that is on the iCub robot. To host the skin, custom-made 3D printed plastic mounts are attached to the robot. The skin is composed of tactile sensors placed on triangular flexible printed circuit boards [17]. Triangles, hosting 10 taxels (the circles) each are assembled into skin patches. The robot is covered with 97 triangles in total, which makes 970 pressure-sensitive taxels over the robot's body. The head is covered with 2 patches with 12 triangles each (240 taxels); every hand is covered with two patches (16 and 8 triangles; 240 taxels per hand); the torso is covered with two patches (14 and 11 triangles; 250 taxels).

### B. CAD-based calibration

One way to acquire 3D coordinates of individual taxels is to combine CAD models with information about the tactile sensors in 2D. Skin triangle centers are placed over the centers of the circular pockets in the plastic part. The CAD models of these 3D printed parts are available and the coordinates and surface normals at the future mounting points of the triangles can be extracted. Accurate 2D information about the coordinates and orientations of every skin triangle and subsequently every taxel within that triangle is available. Fig. 2 provides an overview. Therefore, one can combine these sources of information as follows: 3D positions of mounting points are loaded. Then, the subset of the skin patch with its 2D dimensions is imported into the 3D world as a 2D point cloud—a "sheet of paper" with the skin—and roughly aligned with the 3D part. Then, the skin patch is broken into individual triangles and their centers are aligned with their mounting points in 3D. Their 2D rotation is preserved from the skin patch; the third dimension is set according to the normals of the centers from CAD. Every triangle is thus flat in this representation. The results are shown in Fig. 3 in yellow. The skin for every body part is represented as an individual 3D point cloud.

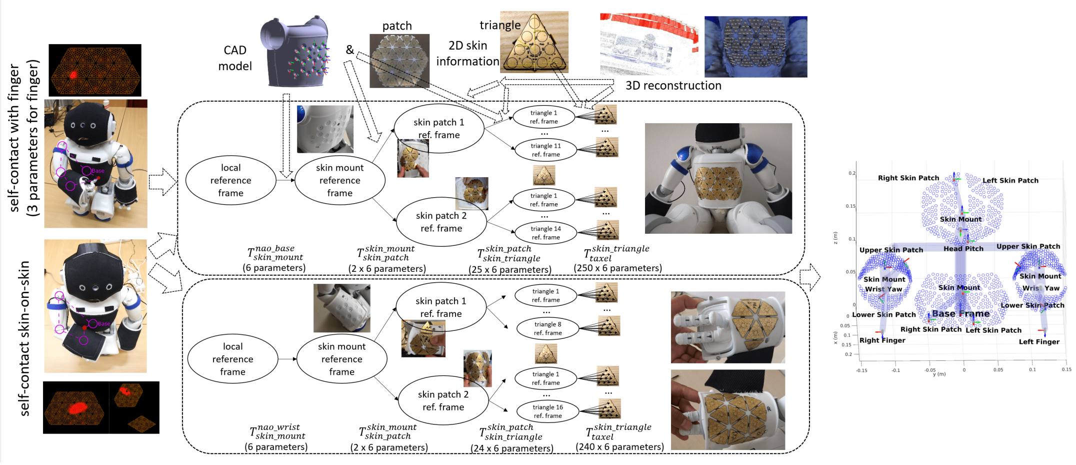


Fig. 2. Overview of the approaches to skin spatial calibration used in this work. See text for details.

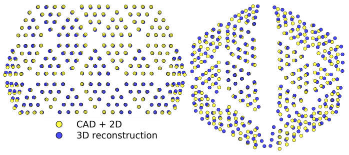


Fig. 3. Skin part calibration – CAD model + 2D skin vs. 3D reconstruction. (Left) Torso. (Right) Head.

This procedure is rather laborious: one needs to keep track of the triangle and taxel IDs from the skin patch and match them with the 3D world. There are the following main sources of error w.r.t. accuracy of the taxel spatial coordinates: (i) Every triangle is flat in this model. However, in reality, every triangle is a flexible PCB and bends to conform to the surface; (ii) The actual placement of the skin does not perfectly match the mounting holes in the plastic part and the triangle orientations from the 2D skin patch are also not perfectly preserved; (iii) The skin for every body part as a 3D point cloud has to be manually aligned with a local reference frame of the robot kinematics.

### C. 3D reconstruction

An overview is provided in Fig. 4. The top layer covering the triangular modules was removed and photos in three different postures (panels A, B, C) taken: a semicircle was traversed with the camera and 125, 78, and 57 photos, respectively, taken (referred to as 3 datasets in this section). In Meshroom, the reconstructed camera positions in one dataset are depicted as red pyramids in panel D; panel E shows the reconstructed textured 3D model. To acquire taxel positions, the model was imported to Meshlab. Individual taxel centers were selected by hand on the model surface (panel F), their 3D coordinates saved along with their taxel IDs (panel G), and imported into Matlab as 3D point clouds (panel H). To determine the scale of the reconstruction, for each dataset, we selected a skin part that was completely visible (torso or head) and measured the distances for 5 pairs of neighboring taxels in every triangle in the model (around 120 measurements in total). Their mean was computed and the real value that was measured manually (6.50 mm; neglecting the effects of curvature between neighboring taxels in 3D) used to set the model scale. The mean difference of every measurement from 6.50 mm over all 3 datasets was 0.086 ± 0.12 mm. Then, the three datasets were brought into the same coordinate system and the mean positions of every taxel computed. For the torso and head, most taxels are present in all three datasets; the mean deviation of the taxel positions reconstructed from individual datasets from their averaged position is 0.15 and 0.22 mm, respectively (with max. deviation of approx. 1 mm). For the hands, for taxels represented in more than one dataset, these deviations were higher (left hand: mean 0.53 mm, max. 2.66 mm; right hand: mean 0.38 mm, max 0.90 mm). Overall, all the errors are sufficiently small compared to the taxel diameter (4.5 mm) or the distance between two taxels (6.5 mm). The mean positions for every taxel from all datasets were then used as the output of 3D reconstruction. Details are in [18].

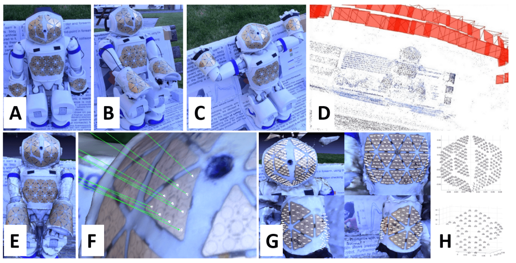


Fig. 4. 3D reconstruction of taxel positions. See text for details.

### D. Estimating Center of Pressure

During self-contact configurations, there are multiple tactile sensors concurrently activated (bottom left of Fig. 2). On average, it is around 14 taxels per contact on one skin part. For calibration, we need the corresponding points that are at the same location in space. Pairwise matching of individual taxels on the two skin parts is error-prone. Therefore, we calculated the *Centre of Pressure (CoP)* of the tactile activations on every skin part. Sometimes large, even discontinuous, areas were activated. Thus, we first clustered the activations and then calculated the CoP for each of them by taking the average 3D position of the taxels belonging to every cluster. With the current model of robot and skin, we matched the positions of the CoPs on the two skin parts in the robot base frame and selected the two closest ones as the contact points $\boldsymbol{X}$ in Equation 1. We used *Mean Shift Clustering* and experimented with different settings of a MaxCentroidDistance parameter (5-50 mm) that sets the maximum distance between taxels and given cluster centroid.

### *E. Custom end effector for self-contact*

As an alternative to self-contact of two skin arrays, we prepared a custom end effector in the shape of a finger or stylus that can generate more localized skin activations (2.3 taxels activated on average; Fig. 2 top left). In this case, the *CoP* is calculated in the same way for the skin part touched by the finger. On the other kinematic chain, the position is given by the tip of the finger. The link connecting the fingertip with the rest of the robot was also added to the parameters being optimized.

### *F. Data collection with self-contact*

For calibration using self-contact, robot joint angles and electronic skin activations need to be recorded simultaneously. To synchronize the data streams, they were recorded on a single PC using a set of Python scripts. The robot was set to the mode in which the joints could be freely rotated and the experimenter manually brought the robot to the self-contact configurations. To make sure that all skin taxels were covered, a visualization was showing online which taxels were already activated in the dataset. The procedure is illustrated in the accompanying video. Robot joint angles were read using the Python API from *NAOqi* running on the robot. Skin activations from every body part were streamed on separate YARP ports containing activations (8bit resolution) of individual taxels. In this work, these pressure values were not used; instead only a list of active taxels for every body part (skin_events port) provided by the skinManager[1] was read. The active taxel list was later synchronized with joint angle data.

The number of contacts collected was: Right Arm – Torso: 1072, Left Arm – Torso: 810, Right Arm – Head: 670, Left Arm – Head: 599, Right Finger – Torso: 1818, Left Finger – Torso: 825, Right Finger – Left Arm: 1814, Left Finger – Right Arm: 908, Left Finger – Head: 893. Note that some of the touches have been discarded in post-processing. The quality of calibration of different areas on the skin depends on how many self-contact data points were collected there. Fig. 5 visualizes this for the skin on the torso (samples from all datasets containing the torso). The color map reveals that the edges of the torso were poorly stimulated, which is mainly because of the robot kinematic constraints.

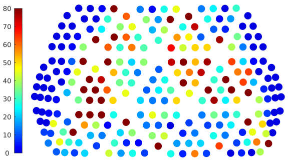


Fig. 5. Number of times a given taxel of the torso was the closest to the calculated CoP on torso skin (MaxNeighborDistance 5 mm).

## IV. Skin spatial calibration – problem formulation

There is a chain of transformations that goes from the last reference frame anchored to the kinematic representation of the robot (*Base frame* for the torso skin; *head pitch* link frame for the head; *wrist yaw* for the hands) to *skin mount*, which is the reference frame of the 3D printed part that hosts the skin, *skin patches* (two skin patches per body part in our case), *triangles* with ref. frame in their centers, and individual *taxels*. This hierarchy respects how the system is physically composed.

Fig. 2 provides an overview of the skin spatial calibration evaluated in this work. The hierarchy of transformations for the skin arrays on the robot torso (upper block in dashed rectangle) and right wrist (bottom) is schematically illustrated. The top row illustrates how information about some of the transformations can be incorporated (shown for the robot torso skin but applies to all skin arrays). CAD models can be used to anchor the skin mounts with respect to the robot kinematic model. Then, together with information about the skin dimensions in 2D, the transformations to every skin patch and every triangle can be approximately set (see Section III-B). The known geometry of individual skin triangles can be used to set the coordinates of individual taxels, assuming the triangle is flat (Section III-B). Information about the positions of individual taxels in 3D can be obtained from 3D reconstruction (Section III-C). This information can be used to set the transforms further up the hierarchy—from the skin mount to patch and from patch to triangle. The information sources described above can be used to acquire taxel 3D coordinates with respect to the robot kinematic model. Note that the schematics illustrate the different options—not all have to be available or used—that will be quantitatively evaluated in this work.

The procedure described above is deterministic and involves no optimization. However, none of this information may be available. Therefore, we complement our calibration approach with self-contact: if the parts in contact are covered with tactile sensors, the contact points at the end of both chains can be identified and their coordinates obtained from forward kinematics. The discrepancy constitutes an error term that can be used for optimization. Two variants are employed in this work: (i) skin-on-skin and (ii) skin to custom end effector (transformation to its tip replaces the need for skin). With the first approach, one can calibrate two skin parts simultaneously, or use one skin part to calibrate another. This approach gives rise to relatively large

[1] Settings: *period 20, minBaseline 3, addThreshold 7, compensationGain 0.2, contactCompensationGain 0.05, zeroUpRawData false, smoothFilter true, smoothFactor 0.5, binarization false*

concurrent activations (see Fig. 2, left; approximately 14 taxels on average) and their centers need to be identified (Section III-D), which introduces errors. Therefore, we tested also the second approach, whereby the custom end effector (Section III-E) on left or right hand touches another body part with skin, thereby calibrating it. The skin activations are more localized and hence more accurate. The link to the finger has to be added to parameters being optimized. The self-contact method can be used alone or in combination with the information from CAD, 2D skin or 3D reconstruction. In this work, the robot kinematics up until the last link (*local reference frame* in Fig. 2) is assumed to be accurate.

Formally, the optimization problem, estimation of the parameter vector $\boldsymbol{\phi}$, can be formulated as follows:

$$\boldsymbol{\phi}^* = \underset{\boldsymbol{\phi}}{\operatorname{argmin}} \sum_{m=1}^{M} ||\boldsymbol{X}_m^A(\boldsymbol{\phi}^A, \boldsymbol{\Theta}_m^A) - \boldsymbol{X}_m^B(\boldsymbol{\phi}^B, \boldsymbol{\Theta}_m^B)||^2, \quad (1)$$

where $M$ is the number of poses used for calibration, and $\boldsymbol{X}_m^A$ and $\boldsymbol{X}_m^B$ are the $m^{\text{th}}$ estimated Cartesian positions of specific points on the robot that are in contact. They are computed using a given parameter estimate $\boldsymbol{\phi}$ and joint angles from joint encoders $\boldsymbol{\Theta}_m$. The positions $\boldsymbol{X}_m$ are expressed in the robot *base frame* located in its waist area and are obtained using forward kinematics (with $\boldsymbol{\Theta}_m$ as input) and the hierarchy of transformations pertaining to the skin (Fig. 2) or the transformation leading to the custom finger end effector.

The parameter vector $\boldsymbol{\phi}$ being optimized may contain: hierarchy of transforms pertaining to skin part $A$ and/or $B$: $X_{skin\_mount}^{nao\_base}$, $X_{skin\_patch}^{skin\_mount}$, $X_{skin\_triangle}^{skin\_patch}$, $X_{taxel}^{skin\_triangle}$ or any subset of them. Additionally, the "finger" transform may be subject to calibration. From the skin parts $A$ and $B$, one has to be the left or right hand and the other may be any other of the three remaining skin parts (torso, head, other hand). The transformations from the Nao *base frame* to the *skin mount* and subsequent transformations to the *skin patch*, *skin triangle*, and *taxel* are general 6D transforms (3D translation, 3D rotation) and hence 6 parameters are needed (we employ the formula of Rodrigues).

*a) Self-contact of two skin arrays:* In this case, $\boldsymbol{X}_m^A$ and $\boldsymbol{X}_m^B$ are determined from the *center of pressure* on the two skin parts (on top of the forward kinematics chain). Note that CoP estimation (Section III-D) depends on the current estimation of the taxel positions. Therefore, this estimation has to be repeated using the list of activated taxels with every calibration step. The details of this method and an extensive set of experiments is available in [16].

*b) Self-contact with custom end effector:* $\boldsymbol{X}_m^A$ is determined from the *CoP* on a skin part and $\boldsymbol{X}_m^B$ from the finger transform—on top of the corresponding forward kinematics chains. The details of this method are available in [18].

### *A. Non-linear least squares optimization*

For solving the calibration task we used the Levenberg-Marquardt algorithm implemented in the non-linear least-squares solver from the Matlab Optimization Toolbox. All experiments were performed using the *Multisensorial robot calibration toolbox* [19] (`https://github.com/ctu-vras/multirobot-calibration`).

### *B. Parameter perturbation and performance evaluation*

To evaluate the dependence of the optimization performance on the quality of the initial estimates of the parameters, we perturbed all parameters being estimated in a given experiment by selecting their values from a uniform distribution centered at the initial estimates $l_i$ (translations) and $\alpha_i$ (rotations). The new initial estimates, $l_i^{new}$, $\alpha_i^{new}$ were given by:

$$\begin{aligned} l_i^{new} &= 100 * uniform[-1;1] + l_i \; [mm], \\ \alpha_i^{new} &= 0.5 * uniform[-1;1] + \alpha_i \; [-]. \end{aligned} \quad (2)$$

Every optimization experiment was run 100 times with different perturbation of the initial parameters and involved batch optimization on a given training set (70% of the dataset selected at random in every run), with an instance of Eq. 1 as the objective function. Each run ended when sum of squares of the points decreased under a given threshold ($1e^{-13}$ in our case) or 100 iterations were reached.

For evaluation, we used the reference positions of every taxel established in Section V-B, with Mean Euclidean Distance (MED) given by following equation:

$$MED = \frac{1}{T} \sum_{t=1}^{T} ||\boldsymbol{X}_t(\boldsymbol{\phi}^*) - \boldsymbol{X}_t(\boldsymbol{\phi}^r)||, \quad (3)$$

where $\boldsymbol{\phi}^*$ is the newly calibrated parameter vector, $\boldsymbol{\phi}^r$ is the reference parameter vector (Section V-B), $T$ is the set of taxels and $\boldsymbol{X}_t(\boldsymbol{\phi})$ is the position of each taxel under the corresponding parameter set ($\boldsymbol{\phi}^*/\boldsymbol{\phi}^r$). The set of taxels may belong to skin part A or B (Eq. 1; or a union of the two if both are calibrated simultaneously) and will contain only those taxels that were subject to perturbation[2]. After obtaining MED for each experiment, the maximum, mean, and Standard Deviation (SD) from all 100 MED values is computed and reported in Section V. No perturbations were applied in Sections V-A and V-G.

## V. Experiments and results

In this section, we present the results of different calibration experiments. No ground truth information regarding the positions of individual taxels with respect to the robot *base frame* is available. However, first, the 3D reconstruction (Section III-C) is sufficiently accurate and can serve as reference for taxel 3D coordinates within a single body part. Second, together, the information from CAD, 3D reconstruction, and both self-contact methods (Fig. 2) is sufficiently rich that the outcome of calibration taking the best of all methods can be taken as reference for the complete hierarchy of reference frames. This will be established in Section V-B. In the remainder, we will report the accuracy of approaches relying on different subsets of the information. Visualization of selected calibration experiments is available in the accompanying video.

[2] Directly or indirectly. That is, if say, the torso skin mount was perturbed, all 250 torso taxels are affected.

| | **IV-A Skin calibration: CAD model + 2D skin vs. 3D reconstruction.** | | |
|---|---|---|---|
| | Max ED [mm] | Mean ED [mm] | SD of ED [mm] |
| Right Arm | 3.49 | 1.23 | 0.64 |
| Left Arm | 6.03 | 1.98 | 0.94 |
| Torso | 2.89 | 0.90 | 0.49 |
| Head | 7.44 | 3.20 | 1.48 |
| | **V-D Skin mount perturbation and calibration** | | |
| Right Arm – skin-on-skin | 2.16 | 1.99 | 0.07 |
| Right Arm – finger | 1.82 | 1.39 | 0.15 |
| Left Arm – skin-on-skin | 2.98 | 2.79 | 0.08 |
| Left Arm – finger | 0.97 | 0.65 | 0.15 |
| Torso – skin-on-skin | 2.12 | 1.97 | 0.05 |
| Torso – finger | 0.65 | 0.51 | 0.06 |
| Head – skin-on-skin | 2.51 | 2.11 | 0.14 |
| | **V-D Mount perturbation and calibration<br>Simultaneous calibration of torso and right arm** | | |
| Right Arm – Torso | 116.79 | 10.38 | 22.61 |
| | **V-E Skin triangle (torso) perturbation and calibration** | | |
| skin-on-skin | 12.33 | 6.67 | 3.04 |
| finger | 6.74 | 2.39 | 1.92 |
| | **V-F Perturbation and calibration of 112 taxels of torso skin** | | |
| skin-on-skin | 18.11 | 12.02 | 2.23 |
| finger | 14.91 | 9.01 | 1.90 |
| combination | 5.22 | 3.11 | 0.65 |
| | **V-G Sequential calibration of skin mount, patch, triangle, and taxel positions with 2D skin part as input (torso)** | | |
| skin-on-skin | 21.52 | 4.04 | 4.89 |
| finger | 21.32 | 3.29 | 4.55 |
| combination | 26.73 | 3.56 | 4.29 |

TABLE I

SUMMARY OF RESULTS FROM SECTIONS V-A AND V-D TO V-G.

### A. *CAD model plus 2D skin geometry*

With CAD models including the sites where skin triangle centers will be approximately placed and the knowledge of skin arrays dimensions in 2D (Section III-B), coordinates of every taxel can be acquired. In this section, we assess the accuracy of this representation by comparing with the taxel coordinates obtained from 3D reconstruction (Section III-C), making the assumption that the taxel coordinates in the local coordinate frame of a body part are correct. The equivalent two point clouds were fitted with use of *SVD* and the algorithm presented in [20]. The 3D positions of all corresponding taxels—their IDs are known—on individual skin parts were compared. The errors are shown in the top part of Table I and visualized in Figure 3. The mean errors are smaller than the taxel diameter (4.5 mm) or the distance between two taxels (6.5 mm). Best match is obtained for the torso, which is the most flat. As expected, the visualizations reveal that the match degrades in areas with bigger curvature, since the "CAD + 2D" method assumes flat triangles. The biggest mismatch is for the head skin array, which can be explained by the difference of the actual placement of the sensors vs. the triangle sites expected from the CAD model—in the bottom center of the head in particular.

### B. *Best calibration (reference)*

To establish the best possible calibration, we proceeded as follows. The torso was taken as the basis. The $T^{base\_frame}_{skin\_mount}$ was taken from the CAD model. The taxel 3D positions from 3D reconstruction were assumed to be correct and then fitted onto the torso using the correspondence of the taxels from 3D reconstruction and their sites in CAD using SVD [20]. The self-contact with finger against the torso was then used to calibrate the finger transform. Then, the skin on the hands was calibrated from self-contact with the torso and self-contact with the other arm with the finger. Then, the head skin hierarchy was calibrated from self-contacts with the hands (skin-on-skin and with finger). Finally, the $T^{base\_frame}_{skin\_mount}$ was unfreezed and let to calibrate using the self-contact datasets.

| | MaxCentroidDistance [mm] | | | | | |
|---|---|---|---|---|---|---|
| | 5 | 10 | 20 | 30 | 40 | 50 |
| Max MED [mm] | 2.44 | 3.61 | 7.47 | 7.82 | 7.66 | 7.69 |
| Mean MED [mm] | 2.22 | 3.32 | 6.98 | 7.37 | 7.27 | 7.25 |
| SD of MED [mm] | 0.08 | 0.12 | 0.15 | 0.17 | 0.16 | 0.16 |

TABLE II

EFFECT OF MAXCENTROIDDISTANCE.

### C. *Experiments with MaxCentroidDistance*

We conducted a series of experiments to investigate the effect of `MaxCentroidDistance` parameter that determines how point contacts needed for calibration are formed from bigger contact areas (Section III-D). Only the skin-on-skin datasets (no "finger") were used. One skin part at a time was calibrated, specifically the $T^{base\_frame}_{skin\_mount}$ (6 parameters), using the other skin part from the dataset. The mount transform of the skin part being calibrated was significantly perturbed, differently in each of 100 optimization runs (Section IV-B and accomp. video). The results using MED (Section IV-B, Eq. 3) are shown in Table II—all values are means from experiments run for the four skin parts separately. The best `MaxCentroidDistance` setting is 5 mm, which will be used in all experiments that follow. A setting of 2mm was also tested, but did not give better results (it is below the diameter of a single taxel).

### D. *Skin mount perturbation and calibration*

Similarly to Section V-C but with `MaxCentroidDistance` fixed to 5 mm, the $T^{base\_frame}_{skin\_mount}$ (6 parameters) of one skin part at a time was calibrated using self-contact datasets, newly adding those with the finger. The results, averages of 100 runs with different perturbations, are in Table I, V-D. For the torso and head, both arms were employed for calibration. For example, "Torso – skin-on-skin" means that Left Arm – Torso and Right Arm – Torso were used. In general, the mean errors are very small (around 2 mm or less) and even maximum errors (of individual taxel positions) are less than 3 mm. Calibration with finger leads to even smaller errors compared to skin-on-skin.

Calibration using self-contact performs best if one kinematic chain is accurately calibrated and used to calibrate another chain. However, this may not always be the case. Here we complement the results above by testing how the method performs if the mounts ($T^{nao\_base}_{torso\_skin\_mount}$, $T^{right\_wrist}_{skin\_mount}$), of two skin parts are perturbed and then calibrated simultaneously (12 parameters in total). The results, Table I, V-D second part, show a significant degradation in performance with max errors of a few cm rather than mm. Sometimes, the optimization fails to align the two skin parts correctly w.r.t. the robot, as demonstrated by the large MED (in one of the 100 runs); median MED was 3.85 mm. Note that the perturbation of both mounts was substantial, with up to 10 cm translation in every axis (Section IV-B).

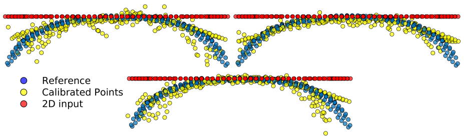


Fig. 6. Sequential calibration of skin mount, patch, triangle, and taxel positions with 2D skin part as input (torso). (Top – left) skin-on-skin, (top – right) finger, (bottom) combination.

### *E. Skin triangle perturbation and calibration (torso)*

We chose two triangles of the torso whose taxels received sufficient stimulation in both skin-on-skin and finger datasets (Fig. 5). Two $T^{skin\_patch}_{skin\_triangle}$ transforms (12 parameters) were thus subject to perturbation and calibration using all the self-contact datasets pertaining to the torso. The results, Table I, V-E, show a significantly better performance of calibration using the finger. For smaller areas of the skin, the skin-on-skin calibration accuracy is limited.

### *F. Skin taxel perturbation and calibration (torso)*

The final layer of the skin hierarchy are the individual sensors: taxels. In this experiment, we perturb the positions of all the torso taxels that received sufficient stimulation (at least 5; see Fig. 5)—112 taxels and hence 672 parameters in total. All datasets pertaining to the torso—left and right hand, skin-on-skin and finger—were used. The results, Table I, V-F, show mean errors of 12 and 9 mm for skin-on-skin and finger calibration, respectively. Employing both methods drives the error down to 3.11 mm.

### *G. Skin taxel calibration with 2D skin part as input (torso)*

In all sections above, with the exception of the perturbed and calibrated parameters, all the remaining parameters of the skin have been set according to the reference values (Section V-B). Here we simulate a different scenario, corresponding to a real use case, whereby we take only 2D information about the skin patches and initialize all taxels' position on a plane in 3D (depicted in red in Fig. 6).

The self-contact datasets involving the torso are used to calibrate all the parameters pertaining to the torso skin. There is a hierarchy of transformations (6 parameters each) that need to be calibrated (see Fig. 2): $X^{nao\_base}_{skin\_mount}$, $2\times$ $X^{skin\_mount}_{skin\_patch}$, $25\times$ $X^{skin\_patch}_{skin\_triangle}$, and $250\times$ $X^{skin\_triangle}_{taxel}$ (1668 parameters in total). Simultaneous calibration of all parameters is possible and gives similar results to those presented here, but the transformations in sequence may compensate for their respective errors, resulting in small errors overall, but numerical instability and incorrect estimation of the individual transforms. In order to guarantee parameter estimation closer to the physical composition of the skin hierarchy, we calibrated sequentially in 4 steps, starting with $X^{nao\_base}_{skin\_mount}$ and finishing with $X^{skin\_triangle}_{taxel}$. The results are visualized in Fig. 6 in yellow and summarized in Table I, V-G. Overall, the calibration was successful in all cases with mean error of 4.04 mm for skin-on-skin, 3.29 mm for finger and 3.56 mm for their combination. Bigger errors are mostly on the sides of the torso, which have been less stimulated (Fig. 5).

| Approach / publication | Mean Error [mm] |
|---|---|
| Albini et al. [14] | $\leq$2.9 (REL) |
| Del Prete et al. [15] | 7.2 (6.6) (REL) |
| Mittendorfer et al. [9] | $\leq$10 (REL) |
| **Our approaches** | |
| 3D reconstruction + self-contact skin-on-skin (torso) | 1.97 |
| 3D reconstruction + self-contact finger (torso) | 0.51 |
| 3D reconstruction + self-contact skin-on-skin (torso and hand simultaneously) | 10.38 |
| taxel perturbation + self-contact both (torso) | 3.11 |
| 2D skin input + self-contact both (torso) | 3.66 |

TABLE III
COMPARISON OF DIFFERENT APPROACHES.

### *H. Summary and comparison to related work*

In Table III we compare our results with others. The comparison is complicated by the fact that the skin, robot, and calibration methods were different and ground truth was often unavailable. In Del Prete et al. [15], Albini et al. [14], and this work, the skin used was very similar, derived from [2], but not identical. Albini et al. [14], employing self-contact with "finger", achieved an error below 2.9 mm. Del Prete et al. [15], estimating from force/torque measurements, reported a mean error of 7.2 mm, which could be reduced to 6.6 mm employing constraints for the robot and skin shape. Mittendorfer et al. [9] employing LEDs on skin as markers achieve errors below 10 mm. We prefer to quote errors for skin patches mounted on the robot, for which errors could often be established in relative terms only (REL). In our case, we assume that the calibration performed in Section V-B is sufficiently accurate to serve as ground truth and the reported errors can be regarded as absolute errors.

In this work, we compared several approaches. Selected results for the torso skin are added to Table III. The first component are 3D positions of taxels within a single skin part. In our case, these can be acquired from 3D reconstruction with very high accuracy (errors of the order of 0.5 mm or less; Section III-C) or from a combination of CAD models and 2D skin geometry with approximately 2 mm error w.r.t. 3D reconstructed positions (Section V-A). The second component is aligning these point clouds of taxel positions in 3D with the robot kinematics. To this end, we employ the self-contact datasets. In Section V-D, we use the taxel positions from 3D reconstruction but significantly perturb their registration w.r.t. the robot (the mount transformation). Calibration is performed using self-contact, with one kinematic chain prepared with reference values (including skin or finger) and used to calibrate the skin on another chain. The mean error of taxel positions is around 2 mm using skin-on-skin and 1 mm using the finger. Simultaneous calibration using self-contact skin-on-skin, with mounts of both skin parts perturbed, leads to significantly larger error for torso and right hand (mean 10.38 mm; median 3.85 mm). If components further down the skin hierarchy—patches, triangle, or taxels—are perturbed, the errors increase. When we perturb all taxels that were sufficiently stimulated in the self-contact datasets (112 taxels of the torso) and calibrate using self-contact, mean errors of 12.02, 9.01, and 3.11 mm are obtained for skin-on-skin, finger, and their combination, respectively. Finally, if only the 2D skin layout is available and self-contact is used for calibration, mean errors of 4.04, 3.29, and 3.66 mm are obtained for skin-on-skin, finger, and their combination, respectively.

## VI. Conclusion and discussion

This work is to our knowledge the first to experimentally compare the accuracy and effort associated with the following skin spatial calibration approaches and their combinations: (i) combining CAD models and skin layout in 2D, (ii) 3D reconstruction from images, (iii) using robot kinematics to calibrate skin using self-contact. In addition, this work is not a proof of concept but deployment of the approaches at scale: the outcome of this work is actual spatial calibration of 970 taxels on the robot body. A detailed summary and comparison with related work is presented in Section V-H.

We used a hierarchical representation of the tactile sensors' position in space respecting how the skin is composed and mounted. We found such a decomposition effective as it allows one to initialize or set different components from different sources depending on the availability of information. For example, 3D information about the layout of individual skin parts may be available from 3D reconstruction and can be inserted. Self-contact can then be employed to align these 3D point clouds with the robot kinematics. This method performed best in our setting leading to errors below the radius of individual sensors. This method can also tolerate if some taxels were not sufficiently stimulated by self-contact.

Calibration using self-contact works well if one chain is known and can be used to calibrate another. If this is not the case, both chains can be calibrated simultaneously, but the accuracy drops. Two variants of self-contact were employed: skin-on-skin and using a custom end effector ("finger"). The former has the advantage that it is completely self-contained—no additional mechanical component is needed—but the activated areas on the skin are bigger which decreases accuracy. The additional end effector needs to be manufactured and calibrated, but can be tailored to the size of the sensors, resulting in more localized activations, and it can be also designed such that it maximizes the areas reachable by self-contact. In our case, self-contact using the finger outperformed skin-on-skin.

For high accuracy of 3D positions of individual sensors, 3D reconstruction provided excellent results. It could be substituted by CAD data combined with 2D skin information, which in our case means every triangle (10 taxels) keeps the 2D layout also in 3D, decreasing the accuracy. However, given that the effort associated with both pipelines is considerable but similar, 3D reconstruction is preferred. In cases where taxels cannot be exposed and made visible or where no information about skin geometry is available (e.g., randomly distributed receptors in [21]), contact or self-contact may remain the only possibility. We have shown that after significant perturbation of the positions of more than 100 taxels, their positions can be calibrated using self-contact. The prerequisite for this, however, is that all the sensors have been sufficiently stimulated.

## References

[1] R. Dahiya, D. Akinwande, and J. S. Chang, "Flexible electronic skin: From humanoids to humans [scanning the issue]," *Proceedings of the IEEE*, vol. 107, no. 10, pp. 2011–2015, 2019.

[2] G. Cannata, M. Maggiali, G. Metta, and G. Sandini, "An embedded artificial skin for humanoid robots," in *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 2008, pp. 434–438.

[3] T. Minato, Y. Yoshikawa, T. Noda, S. Ikemoto, H. Ishiguro, and M. Asada, "Cb2: A child robot with biomimetic body for cognitive developmental robotics," in *Humanoid Robots (Humanoids), IEEE-RAS International Conference on*. IEEE, 2007, pp. 557–562.

[4] Y. Ohmura, Y. Kuniyoshi, and A. Nagakubo, "Conformable and scalable tactile sensor skin for curved surfaces," in *IEEE Int. Conf. on Robotics and Automation*. IEEE, 2006, pp. 1348–1353.

[5] P. Mittendorfer and G. Cheng, "Humanoid multimodal tactile-sensing modules," *IEEE Transactions on Robotics*, vol. 27, no. 3, 2011.

[6] G. Cheng, E. Dean-Leon, F. Bergner, J. R. G. Olvera, Q. Leboutet, and P. Mittendorfer, "A comprehensive realization of robot skin: Sensors, sensing, control, and applications," *Proceedings of the IEEE*, vol. 107, no. 10, pp. 2034–2051, 2019.

[7] J. Hollerbach, W. Khalil, and M. Gautier, "Model identification," in *Springer Handbook of Robotics*, 2nd ed., B. Siciliano and O. Khatib, Eds. Springer, 2016, pp. 113–138.

[8] P. Mittendorfer and G. Cheng, "3D surface reconstruction for robotic body parts with artificial skins," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, Oct 2012.

[9] P. Mittendorfer, E. Dean, and G. Cheng, "3D spatial self-organization of a modular artificial skin," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, Sep. 2014, pp. 3969–3974.

[10] G. Cannata, S. Denei, and F. Mastrogiovanni, "Towards automated self-calibration of robot skin," in *Robotics and Automation (ICRA), IEEE International Conference on*, May 2010, pp. 4849–4854.

[11] Q. Li, R. Haschke, and H. Ritter, "Towards body schema learning using training data acquired by continuous self-touch," in *Humanoid Robots (Humanoids), IEEE-RAS International Conference on*. IEEE, 2015, pp. 1109–1114.

[12] A. Roncone, M. Hoffmann, U. Pattacini, and G. Metta, "Automatic kinematic chain calibration using artificial skin: self-touch in the iCub humanoid robot," in *Robotics and Automation (ICRA), IEEE International Conference on*, 2014, pp. 2305–2312.

[13] K. Stepanova, T. Pajdla, and M. Hoffmann, "Robot self-calibration using multiple kinematic chains – a simulation study on the iCub humanoid robot," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1900–1907, 2019.

[14] A. Albini, S. Denei, and G. Cannata, "Towards autonomous robotic skin spatial calibration: A framework based on vision and self-touch," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*. IEEE, 2017, pp. 153–159.

[15] A. Del Prete, S. Denei, L. Natale, F. M., F. Nori, G. Cannata, and G. Metta, "Skin spatial calibration using force/torque measurements," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, 2011.

[16] L. Rustler, "Artificial skin calibration for the Nao humanoid robot using "self-touch"," Bachelor's thesis, Faculty of Electrical Engineering, Czech Technical University in Prague, 2019.

[17] P. Maiolino, M. Maggiali, G. Cannata, G. Metta, and L. Natale, "A flexible and robust large scale capacitive tactile system for robots," *Sensors Journal, IEEE*, vol. 13, no. 10, pp. 3910–3917, 2013.

[18] B. Potocna, "Artificial skin calibration for a humanoid robot: comparing or combining "self-touch" and 3D reconstruction from images," Bachelor's thesis, Faculty of Electrical Engineering, Czech Technical University in Prague, 2020.

[19] J. Rozlivek, L. Rustler, K. Stepanova, and M. Hoffmann, "Multisensorial robot calibration framework and toolbox," in *Humanoid Robots (Humanoids), IEEE-RAS International Conference on*, 2021.

[20] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-D point sets," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-9, no. 5, pp. 698–700, Sep. 1987.

[21] K. Hosoda, Y. Tada, and M. Asada, "Anthropomorphic robotic soft fingertip with randomly distributed receptors," *Robotics and Autonomous Systems*, vol. 54, no. 2, pp. 104–109, 2006.

# Appendix K

# Automatic self-contained calibration of an industrial dual-arm robot with cameras using self-contact, planar constraints, and self-observation

Stepanova, K., Rozlivek, J., Puciow, F., Krsek, P., Pajdla, T., and Hoffmann, M. (2022). Automatic self-contained calibration of an industrial dual-arm robot with cameras using self-contact, planar constraints, and self-observation. *Robotics and Computer-Integrated Manufacturing* 73: 102250.

Open Access. DOI: `https://doi.org/10.1016/j.rcim.2021.102250`
Youtube video: `https://youtu.be/LMwINqA1t9w`

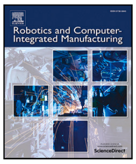
Author contributions: The contribution of M. Hoffmann was 15%.

Robotics and Computer–Integrated Manufacturing 73 (2022) 102250

Contents lists available at ScienceDirect

Robotics and Computer-Integrated Manufacturing

journal homepage: www.elsevier.com/locate/rcm



Full length article

# Automatic self-contained calibration of an industrial dual-arm robot with cameras using self-contact, planar constraints, and self-observation

Karla Stepanova [a,b,1], Jakub Rozlivek [a,1], Frantisek Puciow [a], Pavel Krsek [b], Tomas Pajdla [b], Matej Hoffmann [a,*]

[a] Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic
[b] Czech Institute of Informatics, Robotics, and Cybernetics, Czech Technical University in Prague, Czech Republic

ARTICLE INFO

*Keywords:*
Automatic robot calibration
Self-contained calibration
Self-contact
Self-observation
Self-calibration
Kinematic calibration

ABSTRACT

We present a robot kinematic calibration method that combines complementary calibration approaches: self-contact, planar constraints, and self-observation. We analyze the estimation of the end effector parameters, joint offsets of the manipulators, and calibration of the complete kinematic chain (DH parameters). The results are compared with ground truth measurements provided by a laser tracker. Our main findings are: (1) When applying the complementary calibration approaches in isolation, the self-contact approach yields the best and most stable results. (2) All combinations of more than one approach were always superior to using any single approach in terms of calibration errors and the observability of the estimated parameters. Combining more approaches delivers robot parameters that better generalize to the workspace parts not used for the calibration. (3) Sequential calibration, i.e. calibrating cameras first and then robot kinematics, is more effective than simultaneous calibration of all parameters. In real experiments, we employ two industrial manipulators mounted on a common base. The manipulators are equipped with force/torque sensors at their wrists, with two cameras attached to the robot base, and with special end effectors with fiducial markers. We collect a new comprehensive dataset for robot kinematic calibration and make it publicly available. The dataset and its analysis provide quantitative and qualitative insights that go beyond the specific manipulators used in this work and apply to self-contained robot kinematic calibration in general.

## 1. Introduction

Accurate calibration is essential for the performance of every robot. Traditional calibration procedures involve some form of external measuring apparatus and become impractical if the robot itself or the site where it is deployed change frequently—the current trend in automation with the shift from mass to small batch production. Furthermore, collaborative and social robots often employ cheaper and more elastic materials, making them less accurate and prone to frequent re-calibration. At the same time, advances in sensor technology make affordable but increasingly accurate devices such as standard or RGB-D cameras, tactile, force, or inertial sensors available; often, robots come already equipped with a selection of them. These factors together constitute the need and the opportunity to perform an automated self-contained calibration relying on redundant information in these sensory streams originating from the robot platform itself.

We present a robot kinematic calibration method, which combines complementary calibration approaches: self-contact, planar constraints, and self-observation. There are two main approaches to robot calibration [1]: (i) *open-loop* approaches, where the manipulator is not in mechanical (physical) contact with the environment and an external metrology system is used to measure robot pose components, and (ii) *closed-loop* approaches, where physical constraints on robot pose components substitute for external measurements. However, in their standard formulations, both approaches require a setup that is external to the robot itself. This work aims to extend these well-established tools to *automatic self-contained robot calibration* [2] to include camera reprojection errors, constraints arising from robot self-contact, or simultaneous calibration of multiple kinematic chains.

In this work, we employ two industrial manipulators mounted on a common base with force/torque sensors at their wrists, two cameras overlooking the workspace, and special end effectors with fiducial markers. Using this setup, we study different kinematic calibration methods: self-contact, planar constraints, and self-observation. These can be employed separately or in combination (simultaneously or

[*] Corresponding author.
*E-mail address:* matej.hoffmann@fel.cvut.cz (M. Hoffmann).
[1] Both authors contributed equally.

https://doi.org/10.1016/j.rcim.2021.102250
Received 24 August 2020; Received in revised form 27 May 2021; Accepted 20 August 2021

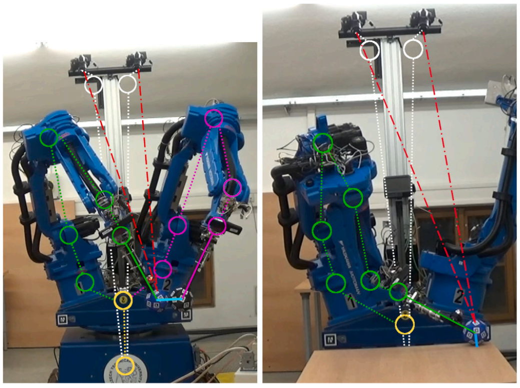


**Fig. 1.** Setups for automatic calibration of the dual-arm robot with graphs of the kinematic chains. Setup for self-touch calibration (left) and calibration when a horizontal plane is touched (right). All chains originate in a common base frame (bottom yellow circle). The left and right arm chains are drawn in purple and green, respectively. The eye chains are drawn in white. Red lines denote reprojection into the cameras. Cyan indicates the distance between end effector centers (one diameter) or between the end effector and a plane (end effector radius). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

sequentially). Fig. 1 shows an example: a self-contact configuration provides constraints that can be exploited to calibrate the kinematic chain of one or both arms. At the same time, the end effectors are observed by the cameras, providing additional data (equations) for calibrating the manipulators' kinematics and the camera orientation parameters.

We evaluate the performance of individual approaches on the calibration of: (i) parameters of the custom end effector, (ii) joint offsets of the complete kinematic chain of one arm, and (iii) the full Denavit-Hartenberg (DH) representation of the platform. Calibration using an external measurement device (laser tracker) is performed and serves together with the nominal manipulators' kinematic parameters as a reference. The calibration performance of individual methods is compared on an independent testing dataset which covers a significant portion of the robot workspace.

We present four main contributions. First, we provide a thorough experimental comparison of geometric kinematic calibration using self-contained approaches—self-contact, planar constraints, and self-observation—with kinematic calibration based on an external laser tracker. We demonstrate the viability of the self-contained approaches. Second, we design ways how the individual methods can be combined into a single cost function. The merits of such a synergistic approach leading to better identifiability of the calibrated parameters and generalization to the part of the robot workspace, which was not used for parameter calibration, are empirically evaluated. Third, we compare simultaneous (all parameters at once) with sequential calibration. Fourth, we collect a new comprehensive dataset for robot kinematic calibration and make it publicly available. This dataset and its analysis provide quantitative and qualitative insights that go beyond the specific manipulators used in this work and apply to self-contained robot kinematic calibration in general.

To our knowledge, this work is the first that bridges the realm of very accurate industrial calibration using external metrology with compact approaches relying on sensors on the robot—typical for humanoid or social robotics. The innovation this work brings is that it expands the portfolio of traditional calibration methods by self-contained approaches such as self-contact or self-observation. While traditionally industrial robots did not have means to detect contacts, this is changing now with the advent of collaborative robots where sensitive contact detection is necessary for safe human–robot collaboration. Furthermore, torque sensing within the robot structure or joint/torque sensors at the flange facilitate dexterous manipulation and interaction with the environment. At the same time, integrated vision sensors (cameras, RGB-D sensors) are becoming popular. Here we provide a framework that makes it possible to exploit all available calibration methods and sensory streams simultaneously and combine them in an optimal way, including weighting by their respective accuracy.

This article is structured as follows. Related work is reviewed in Section 2. Section 3 describes Experimental setup and data acquisition: Robot (Section 3.1) and Laser tracker (Section 3.2) setup, robot control using force feedback (Section 3.3), robot and end effector dimensions (Section 3.4), camera calibration (Section 3.5), and description of individual acquired datasets (Section 3.6). The optimization problem formulation for different combinations of kinematic chains is described in Section 4. Experimental results for end effector, manipulator offsets, and all DH parameters calibration (including laser tracker reference) are presented in Section 5. Finally, we present the conclusion (Section 6) and discussion, including future work (Section 7). The accompanying video illustrating the experiments is available at https://youtu.be/LMwINqA1t9w; the dataset is available from [3].

## 2. Related work

Standard robot kinematic calibration employs different mechanical systems—for example, measurement arms [4]—or contactless measurement systems like laser trackers [5–8]. Cameras are used less often: for example, stereo cameras combined with a calibration sphere [9] or a camera together with a laser pointer at the end effector [10]. All of these systems require specialized external equipment. Our platform was previously calibrated using two different methods: (1) Redundant parallel Calibration and measuring Machine (RedCaM) by Beneš et al. [11], Volech et al. [12], and (2) Leica laser tracker. Petrík and Smutný [13] reviewed the precision of these methods using a linear probe sensor.

Since we aim at automatic self-contained multisensorial calibration—that is, calibration using sensors on the robot, involving multiple sensory modalities like vision and touch—we next focus on reviewing the state of the art in calibration methods that do not rely on external metrology systems. We will also pay special attention to humanoid-like setups that offer the richest possibilities for self-contained calibration.

*Calibration by self-observation.* Cameras mounted on a robot can be used to calibrate the robot by closing the calibration loop through self-observation of its end effectors. The theory for this approach is laid out in [14] for a stereo camera system observing a robot arm. The manipulator's kinematics and extrinsic and intrinsic camera parameters are calibrated. Self-observation has been applied to humanoid robots viewing their hands with fiducial markers using online methods to calibrate kinematics relying on gradient descent by Hersch et al. [15] and recursive least squares estimation by Martinez-Cantin et al. [16]. Fiducial markers can sometimes be avoided when the robot wrist, hand, or fingertip are identified directly in the image [17,18]. The work of Birbach et al. [2] on the humanoid Agile Justin will be discussed in more detail below.

*Calibration using physical constraints.* The next family of approaches exploits physical contacts of the end effector with the environment, such as fixing the end effector to the ground [19] or using more complex setups [20–22]. Some form of force sensing on the part of the manipulator is required. Kinematic calibration using plane constraints (with known or unknown parameters of the plane) was explored by Ikits and Hollerbach [23]; they proposed a new approach focusing on the proper definition of the base and end link frames and evaluated primarily in simulation. Zhuang et al. [24] explored multiple variants of plane constraints and the option with/without known plane parameters and demonstrated their results on a PUMA 560 robot. In particular, they showed that a single-plane constraint does not necessarily guarantee that all kinematic parameters of the robot will be observable. On the other hand, a multiple-plane constraint should be a remedy to this problem. They show that the data collected from 3 planar constraints is equivalent to the data collected from a point measurement device provided that: (1) all three planes are mutually non-parallel; (2) the identification Jacobian of the unconstrained system is nonsingular; and (3) the measured points from each individual plane do not lie on a line on that plane. Joubair and Bonev [25] showed how multi-planar constraints can significantly improve the accuracy of calibration of a six-axis serial robot. Zenha et al. [26] had the simulated iCub humanoid touch three known planes, employing adaptive parameter estimation (Extended Kalman Filter) for kinematic calibration. Khusainov et al. [27] exploit a specific type of mechanical coupling as they fix the end effector of a manipulator to the legs of a humanoid robot that is being calibrated. A point and distance constraint can also be obtained from visual sensing [28].

*Calibration by self-contact (self-touch).* Self-contact constitutes a specific, less common way of kinematic loop closure available only to humanoid-like or dual arm setups. Additionally, corresponding sensory and motor equipment such that this self-contact can be performed in a controlled manner is needed. One possibility is to utilize artificial electronic skins covering specific areas or complete robot bodies (see [29,30] for recent overviews). A tactile array may be used for contact detection. If accurate spatial calibration of the skin is available, then additional components of the self-contact configuration—where contact occurs on each of the two intersecting chains—can be measured. Roncone et al. [31] performed kinematic calibration on the iCub using autonomous self-touch—index finger on the contralateral forearm; Li et al. [32] employed a dual KUKA arm setup with a sensorized "finger" and a tactile array on the other manipulator. Mittendorfer and Cheng [33] also exploit artificial skin to learn models of robots, but their methods primarily utilize the signals from accelerometers embedded in their multimodal skin. Alternatively, if controlled self-contact can be established but the exact position is not measured—typically when using force/torque sensing—such constraints can also be employed for calibration, as will be demonstrated in this work.

*Self-contained multisensorial calibration.* There are only a few approaches that exploit "multisensorial" (or "multimodal") input for self-contained calibration. Birbach et al. [2] calibrated the humanoid robot Justin observing its wrist. Sensors were fused by minimizing a single cost function that aggregates the errors obtained by comparing the discrepancies between simulated projections (left and right camera images, Kinect image, Kinect disparity) and the wrist position from forward kinematics. An inertial term from an IMU in the head was also considered. It is claimed that while pair-wise calibration can lead to inconsistencies, calibrating everything together in a "mutually supportive way" is the most efficient. Limoyo et al. [34] used contact constraints from sliding on a surface together with RGB-D camera information to formulate a self-calibration problem for a mobile manipulator to estimate camera extrinsic parameters and manipulator joint angle biases. The former part is also experimentally verified. Stepanova et al. [35] systematically studied on the simulated iCub humanoid robot how self-observation, self-contact, and their combination can be used for self-calibration and evaluated the relative performance of these approaches by varying the number of poses, initial parameter perturbation and measurement noise. They found that employing multiple kinematic chains ("self-observation" and "self-touch") is superior in terms of optimization results as well as observability.

Largely orthogonal to the calibration types mentioned above, observability and identifiability of the system and speed of optimization convergence can be improved by (i) combination of geometric and parametric approaches to kinematic identification [36], (ii) improving the error model by incorporating impact of strain wave gearing errors [37] and unifying various error sources [38], or (iii) improving the optimization method [39].

In summary, existing works on self-contained automatic calibration have typically focused on a single approach—relying on self-observation, physical constraints, or self-contact. Birbach et al. [2] combined multiple sensors. However, first, essentially only "self-observation" chains closed through different vision-like sensors in the robot head were used. Second, only the case where all chains were combined using a single cost function was considered. In [34], self-observation and contact information are combined, but the results have a proof-of-concept character. Our work is inspired by the simulation study of Stepanova et al. [35] but presents a completely new setting and results on a different platform. Using a dual-arm industrial robot with force/torque sensing and cameras, we present several formulations of the optimization problem for geometric kinematic calibration and empirical verification using (i) self-contact, (ii) planar constraints, and (iii) self-observation. The pros and cons of the individual methods and their synergies are assessed and compared to an independent calibration using an industrial quality laser tracker.

## 3. Experimental setup and data acquisition

In this section, we introduce the experimental setup: the robot platform and its control, dimensions, and camera calibration. Then we present the process of data acquisition and the structure of the collected datasets.

### 3.1. Robot setup description

For our experiments, we used a robotic platform developed in the CloPeMa project [40] – Figs. 1 and 2. It consisted of two industrial manipulators Yaskawa-Motoman MA1400 installed on top of a Yaskawa R750 robotic turntable, which allows rotation of the two manipulators around the vertical axis, a control unit, and two computers connected to a local network. Two Nikon D3100 cameras with Nikkor 18-55 AF-S DX VR lens were mounted side-by-side on the turntable over the robot base, moving along with the turntable.

Manipulators were equipped with ATI Industrial Automation Mini45 6-axis force/torque (F/T) sensors, placed between the last link of the manipulator and the end effector. All the different parts of the robot system were integrated into and operated with Robot Operating System (ROS) [41]. MoveIt! planning framework was used to control the robot.

We further equipped the robot with custom-made end effectors, one on each robot arm, that can be used to achieve self-contact and, at the same time, visual self-observation by the two cameras. To this end, we designed and 3D printed icosahedron shapes (Fig. 3 (left)) that have 20 flat faces, where fiducial markers can be placed, interleaved with 10 spherical surface regions that can be used for self-contact. The end effectors (further also referred to as icosahedron) were then attached to a steel rod. The collision model (Fig. 3, center) for the end effector was added to the robot model. For calibration employing self-contact or contact with a plane, the end effectors are treated as spheres (see Fig. 3, center). The spherical surfaces with corresponding reference frames are shown in Fig. 3, right.

For visual self-observation, having 20 evenly spaced ArUco markers should ensure that at least 3 of them are always seen by each camera unless another part of the robot occludes the view. We used the OpenCV ArUco module to detect the markers in images.

### 3.2. Laser tracker setup

For acquiring reference parameter values and a baseline for self-contained calibration, we used an external "laser tracker", i.e. Leica Geosystems Absolute Tracker AT402, which is a portable 3D absolute tracker for fully guided measurement processes [42]. The tracker collects the 3D coordinates of points in the coordinate system of the tracker. The measurement has a resolution of 0.1 μm. The tracker was

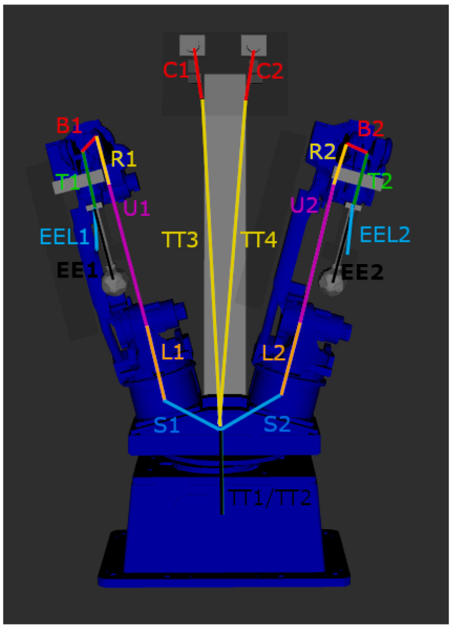


**Fig. 2.** Setup of the whole dual arm robot: visualization of individual links and their location on the robot. DH parameters of these links are listed in Table 1.

placed approximately 4 m from the manipulator. Thus, the typical $U_{xyz}$ uncertainty of the measurement was $\pm 7.5 + 3 \times 4$ microns [42,43].

The retroreflector was attached to the robot approximately 25 cm from the last joint. During data collection, the distance from the laser tracker was between 2.5 and 4.5 m. The retroreflector and its collision model are shown in Fig. 3 (left and center).

### 3.3. Robot control for contact configurations using force feedback

Driving the two manipulator arms into physical contact is necessary for exploiting self-contact in robot calibration. A point in the workspace was chosen where the end effectors should touch. Then, the configuration of each arm and the movement trajectory were obtained using MoveIt!. Each contact consists of three or four phases depending on the experiment. In the first phase, the robot right arm moves at a speed of 0.7 $\mathrm{m\,s^{-1}}$ to a point close to the desired pose. In the self-contact experiment, this is followed by an analogous movement of the left arm. Then the right arm starts moving to an anticipated contact point (in fact, a small negative distance was used) with the left arm or a plane at a speed of 0.1 $\mathrm{m\,s^{-1}}$ until the collision is detected by F/T sensors. The contact thresholds were determined empirically. Once the end effector gets into contact, the arm stops moving and cameras are triggered to take photographs. Joint angles are recorded. Afterwards, the right arm is slowly (0.1 $\mathrm{m\,s^{-1}}$) moving to the former position (from the end of the first phase). Then the new target is selected and the whole process is repeated. For laser tracker experiments, the deceleration phase is skipped and the robot arm chosen moves directly at a speed of 0.7 $\mathrm{m\,s^{-1}}$ in free space to predefined configurations which were sampled in a way that a significant part of the robot joint space is covered.

### 3.4. Robot and end effector dimensions and representation

A parametric model of the whole robot including the custom-made end effectors was created. The robot's complete kinematic model was described by Denavit-Hartenberg (DH) notation, including cameras, end effectors, and laser tracker retroreflector.

**Initialization of parameters.** To obtain the initial model, we used the nominal parameters of the robot provided by the manufacturer, previously acquired transformation matrices describing the mounting of the robot on the base, CAD model and manual measurements of the custom end effector and laser tracker retroreflector placement (the link EEL1/2 (7b) represents the transformation to the retroreflector from the last robot joint — see Figs. 2, 3, and Table 1), and manual measurement of the cameras' attachment. Using the Denavit-Hartenberg (DH) notation [44], Table 1 shows parameters of the manipulators, including the custom end effector or the retroreflector placement. Table 2 shows the two links to every camera.

The manipulator has 6 actuated rotational joints, denoted $S$, $L$, $U$, $R$, $B$ and $T$ (*in italics*), connected by links which are denoted S, L, U, R, B, and T in the order from the turntable to the end effector. Joint $S$ connects the turntable with link S of the robot; joint $L$ connects link S with link L, and so on — see Fig. 2. We added one last link to every manipulator's kinematic chain. These links represent the transformation to the end effector (icosahedron) (EE1, EE2) or retroreflector (EEL1, EEL2). Links between the origin and the turntables (TT1 and TT2) were added as well. The camera chain was also expressed using DH representation and is composed of 2 links: the link between the origin and the camera turntable (TT3, TT4) and the link to the camera entrance pupil (C1, C2). These last links (EE1, EE2, EEL1, EEL2, C1, C2) are not connected by an actual joint, and thus their joint angle was set to zero. These joints may still have a non-zero offset $o$ though.

We consider the complete kinematic system of the robot platform to be composed of four individual kinematic chains described by the DH parameters: (1) Right arm (TT1, S1, L1, U1, R1, B1, T1, EE1), (2) Left arm (TT2, S2, L2, U2, R2, B2, T2, EE2), (3) Right camera (TT3, C1) and (4) Left camera (TT4, C2). Additionally, we consider kinematic chains for laser tracker measurement (5) Right arm (TT1, S1, L1, U1, R1, B1, T1, EEL1), and (6) Left arm (TT2, S2, L2, U2, R2, B2, T2, EEL2). All the kinematic chains start in the base frame. The transformation from the base frame to the first joint, i.e. to the turntable joint, is identity. For calibration using the laser tracker, we replaced the icosahedron end effector links with links ending in the retroreflectors (EEL1, EEL2) (see Fig. 2, Table 1 for robot DH parameters, and Table 2 for camera DH

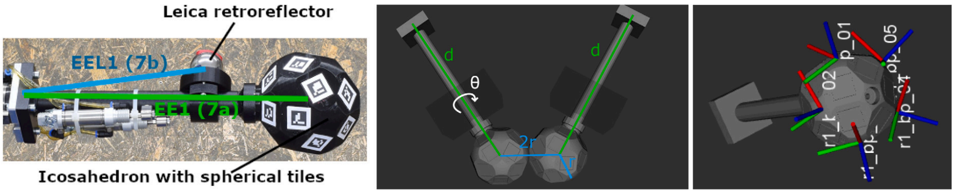


**Fig. 3.** Custom end effector with icosahedron and laser tracker retroreflector (left), its collision model in self-contact configuration ($d$ and $o$ are DH parameters of end effector and $r$ is the radius of icosahedrons) (center), and spherical tiles used for contacts with their reference frames (right).

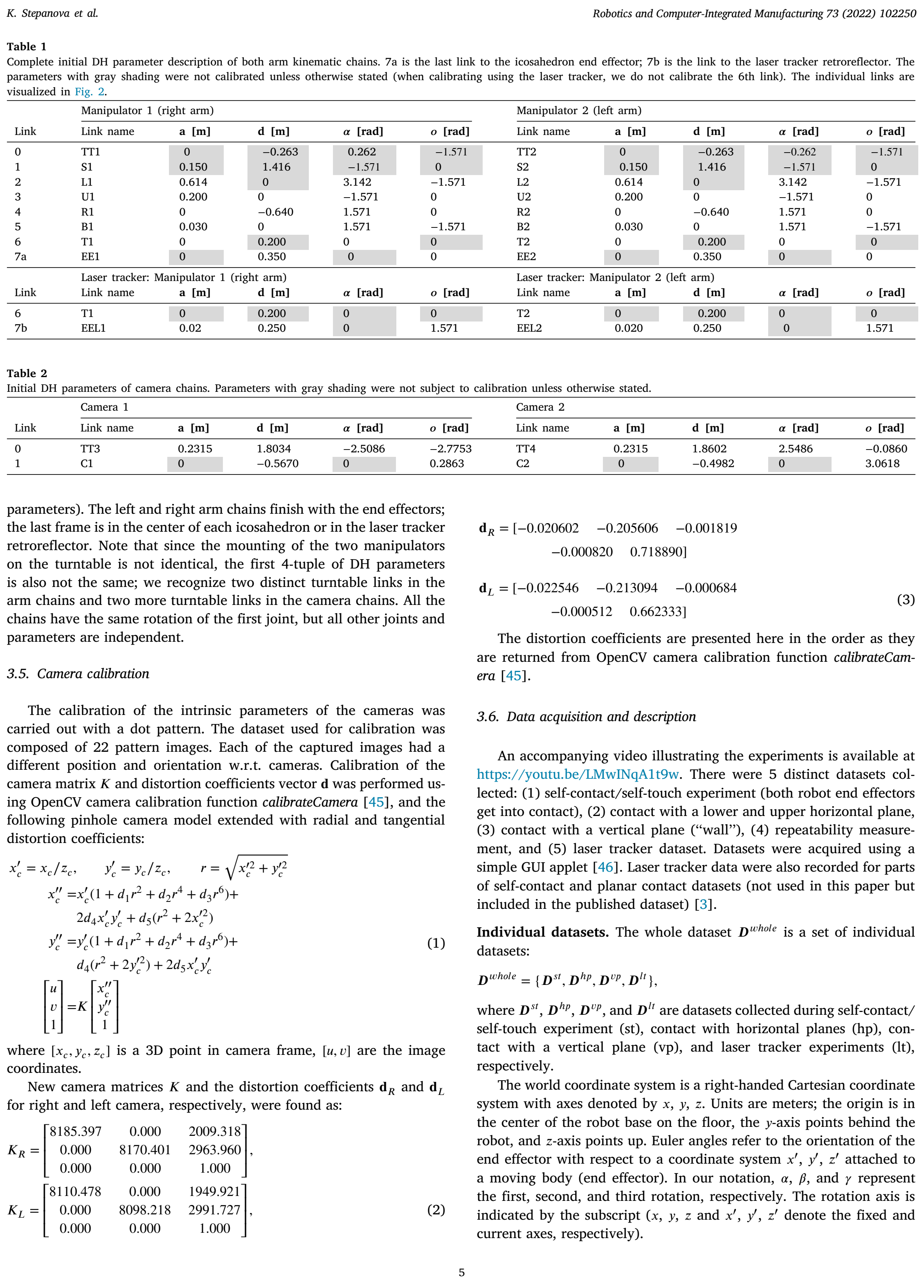


**Table 1**
Complete initial DH parameter description of both arm kinematic chains. 7a is the last link to the icosahedron end effector; 7b is the link to the laser tracker retroreflector. The parameters with gray shading were not calibrated unless otherwise stated (when calibrating using the laser tracker, we do not calibrate the 6th link). The individual links are visualized in Fig. 2.

| | Manipulator 1 (right arm) | | | | | Manipulator 2 (left arm) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Link | Link name | **a [m]** | **d [m]** | **$\alpha$ [rad]** | **$o$ [rad]** | Link name | **a [m]** | **d [m]** | **$\alpha$ [rad]** | **$o$ [rad]** |
| 0 | TT1 | 0 | −0.263 | 0.262 | −1.571 | TT2 | 0 | −0.263 | −0.262 | −1.571 |
| 1 | S1 | 0.150 | 1.416 | −1.571 | 0 | S2 | 0.150 | 1.416 | −1.571 | 0 |
| 2 | L1 | 0.614 | 0 | 3.142 | −1.571 | L2 | 0.614 | 0 | 3.142 | −1.571 |
| 3 | U1 | 0.200 | 0 | −1.571 | 0 | U2 | 0.200 | 0 | −1.571 | 0 |
| 4 | R1 | 0 | −0.640 | 1.571 | 0 | R2 | 0 | −0.640 | 1.571 | 0 |
| 5 | B1 | 0.030 | 0 | 1.571 | −1.571 | B2 | 0.030 | 0 | 1.571 | −1.571 |
| 6 | T1 | 0 | 0.200 | 0 | 0 | T2 | 0 | 0.200 | 0 | 0 |
| 7a | EE1 | 0 | 0.350 | 0 | 0 | EE2 | 0 | 0.350 | 0 | 0 |
| | Laser tracker: Manipulator 1 (right arm) | | | | | Laser tracker: Manipulator 2 (left arm) | | | | |
| Link | Link name | **a [m]** | **d [m]** | **$\alpha$ [rad]** | **$o$ [rad]** | Link name | **a [m]** | **d [m]** | **$\alpha$ [rad]** | **$o$ [rad]** |
| 6 | T1 | 0 | 0.200 | 0 | 0 | T2 | 0 | 0.200 | 0 | 0 |
| 7b | EEL1 | 0.02 | 0.250 | 0 | 1.571 | EEL2 | 0.020 | 0.250 | 0 | 1.571 |

**Table 2**
Initial DH parameters of camera chains. Parameters with gray shading were not subject to calibration unless otherwise stated.

| | Camera 1 | | | | | Camera 2 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Link | Link name | **a [m]** | **d [m]** | **$\alpha$ [rad]** | **$o$ [rad]** | Link name | **a [m]** | **d [m]** | **$\alpha$ [rad]** | **$o$ [rad]** |
| 0 | TT3 | 0.2315 | 1.8034 | −2.5086 | −2.7753 | TT4 | 0.2315 | 1.8602 | 2.5486 | −0.0860 |
| 1 | C1 | 0 | −0.5670 | 0 | 0.2863 | C2 | 0 | −0.4982 | 0 | 3.0618 |

parameters). The left and right arm chains finish with the end effectors; the last frame is in the center of each icosahedron or in the laser tracker retroreflector. Note that since the mounting of the two manipulators on the turntable is not identical, the first 4-tuple of DH parameters is also not the same; we recognize two distinct turntable links in the arm chains and two more turntable links in the camera chains. All the chains have the same rotation of the first joint, but all other joints and parameters are independent.

### 3.5. Camera calibration

The calibration of the intrinsic parameters of the cameras was carried out with a dot pattern. The dataset used for calibration was composed of 22 pattern images. Each of the captured images had a different position and orientation w.r.t. cameras. Calibration of the camera matrix $K$ and distortion coefficients vector $\mathbf{d}$ was performed using OpenCV camera calibration function *calibrateCamera* [45], and the following pinhole camera model extended with radial and tangential distortion coefficients:

$$x'_c = x_c/z_c, \qquad y'_c = y_c/z_c, \qquad r = \sqrt{x'^2_c + y'^2_c}$$
$$\begin{aligned} x''_c =& x'_c(1 + d_1 r^2 + d_2 r^4 + d_3 r^6) + \\ & 2d_4 x'_c y'_c + d_5(r^2 + 2x'^2_c) \\ y''_c =& y'_c(1 + d_1 r^2 + d_2 r^4 + d_3 r^6) + \\ & d_4(r^2 + 2y'^2_c) + 2d_5 x'_c y'_c \end{aligned} \tag{1}$$
$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \begin{bmatrix} x''_c \\ y''_c \\ 1 \end{bmatrix}$$

where $[x_c, y_c, z_c]$ is a 3D point in camera frame, $[u, v]$ are the image coordinates.

New camera matrices $K$ and the distortion coefficients $\mathbf{d}_R$ and $\mathbf{d}_L$ for right and left camera, respectively, were found as:

$$K_R = \begin{bmatrix} 8185.397 & 0.000 & 2009.318 \\ 0.000 & 8170.401 & 2963.960 \\ 0.000 & 0.000 & 1.000 \end{bmatrix},$$
$$K_L = \begin{bmatrix} 8110.478 & 0.000 & 1949.921 \\ 0.000 & 8098.218 & 2991.727 \\ 0.000 & 0.000 & 1.000 \end{bmatrix}, \tag{2}$$

$$\mathbf{d}_R = [-0.020602 \quad -0.205606 \quad -0.001819 \quad -0.000820 \quad 0.718890]$$
$$\mathbf{d}_L = [-0.022546 \quad -0.213094 \quad -0.000684 \quad -0.000512 \quad 0.662333] \tag{3}$$

The distortion coefficients are presented here in the order as they are returned from OpenCV camera calibration function *calibrateCamera* [45].

### 3.6. Data acquisition and description

An accompanying video illustrating the experiments is available at https://youtu.be/LMwINqA1t9w. There were 5 distinct datasets collected: (1) self-contact/self-touch experiment (both robot end effectors get into contact), (2) contact with a lower and upper horizontal plane, (3) contact with a vertical plane ("wall"), (4) repeatability measurement, and (5) laser tracker dataset. Datasets were acquired using a simple GUI applet [46]. Laser tracker data were also recorded for parts of self-contact and planar contact datasets (not used in this paper but included in the published dataset) [3].

**Individual datasets.** The whole dataset $\boldsymbol{D}^{whole}$ is a set of individual datasets:

$$\boldsymbol{D}^{whole} = \{\boldsymbol{D}^{st}, \boldsymbol{D}^{hp}, \boldsymbol{D}^{vp}, \boldsymbol{D}^{lt}\},$$

where $\boldsymbol{D}^{st}$, $\boldsymbol{D}^{hp}$, $\boldsymbol{D}^{vp}$, and $\boldsymbol{D}^{lt}$ are datasets collected during self-contact/self-touch experiment (st), contact with horizontal planes (hp), contact with a vertical plane (vp), and laser tracker experiments (lt), respectively.

The world coordinate system is a right-handed Cartesian coordinate system with axes denoted by $x$, $y$, $z$. Units are meters; the origin is in the center of the robot base on the floor, the $y$-axis points behind the robot, and $z$-axis points up. Euler angles refer to the orientation of the end effector with respect to a coordinate system $x'$, $y'$, $z'$ attached to a moving body (end effector). In our notation, $\alpha$, $\beta$, and $\gamma$ represent the first, second, and third rotation, respectively. The rotation axis is indicated by the subscript ($x$, $y$, $z$ and $x'$, $y'$, $z'$ denote the fixed and current axes, respectively).

**Table 3**
Dataset overview.

| Dataset | | Contact positions | Orientations | No. datapoints |
|---|---|---|---|---|
| Self-touch ($\boldsymbol{D}^{st}$) | | $4 \times 4 \times 2$ grid in the $xyz$-box with $x \in [-0.3, 0.2]$m, $y \in [-1.1, -0.6]$m and $z = \{0.8, 1\}$m | 9 combinations of right and left arm orientations — see Fig. 4 | 566 datapoints (10 not logged) |
| Horizontal planes ($\boldsymbol{D}^{hp}$) | | $5 \times 5$ grid in the $xy$-plane with $x \in [-0.4, 0.3]$m, $y \in [-1.35, -0.65]$m, and $z = \{0.67, 85\}$m | 5 orientations in Euler angles $(\alpha_x, \beta_y, \gamma_z)$: $\{(0, 185, 52), (0, 180, -20), (0, 180, 124), (0, 180, -164), (0, 180, -92)\}$ | 454 datapoints (48 not logged) |
| Vertical plane ($\boldsymbol{D}^{vp}$) | | $5 \times 5$ grid in the $yz$-plane with $y \in [-1.2, -0.7]$ m, $z = [0.7, 1.1]$ m, and $x = 0.05$ m | 5 orientations in Euler angles $(\alpha_z, \beta_y, \gamma_z)$: $\{(180, 90, 0), (108, 90, 0), (36, 90, 0), (-36, 90, 0), (-108, 90, 0)\}$ | 248 datapoints (2 not logged) |
| Repeatability measurement | st | $[x, y, z]^T = [0.225, -0.85, 0.8]^T$m<br>$[x, y, z]^T = [-0.025, -0.85, 1]^T$m | right arm $(\alpha_z, \beta_y, \gamma_z) : (-108, 90, 0)$<br>left arm $(\alpha_z, \beta_y, \gamma_z) : (0, -90, 0)$ | 20 datapoints |
| | hp | $[x, y, z]^T = [0.3, -0.825, 0.67]^T$m<br>$[x, y, z]^T = [-0.4, -0.825, 0.67]^T$m | right arm $(\alpha_x, \beta_y, \gamma_z) : (0, 185, 52)$ | 20 datapoints |
| | vp | $[x, y, z]^T = [0.17, -0.825, 1.1]^T$m<br>$[x, y, z]^T = [0.17, -0.95, 0.7]^T$m | right arm $(\alpha_z, \beta_y, \gamma_z) : (-108, 90, 0)$ | 20 datapoints |
| Laser tracker ($\boldsymbol{D}^{lt}$) | | whole configuration space of the manipulator was sampled (see text for details) | – | 586 datapoints (99 not logged) |

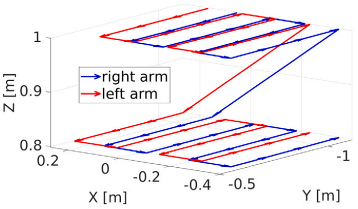


**Fig. 4.** Positions of end effector centers during the self-contact experiment. The end of each arrow denotes a position of end effector center in individual poses (red — left arm, blue — right arm). The distance between left and right arm position is given by two times the radius of the end effector in contact.

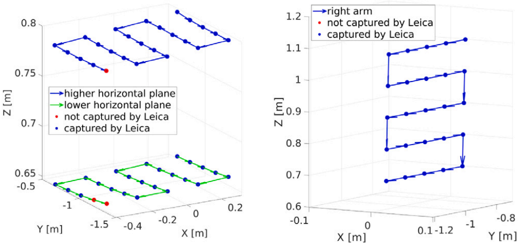


**Fig. 5.** Positions of end effector centers in the horizontal plane contact datasets (left, higher plane in blue, lower plane in green) and vertical plane contact dataset (right) with the information whether the pose was logged by laser tracker (blue) or not (red).

Contact points of end effectors in datasets $\boldsymbol{D}^{st}$, $\boldsymbol{D}^{hp}$, and $\boldsymbol{D}^{vp}$ were planned on grids in the manipulators' workspace, as shown in Figs. 4 and 5 for the self-contact and contact with planes, respectively. Several end effector orientations were tested for every position. A detailed description of individual datasets with contact points and orientations is provided in Table 3. Every experiment was repeated twice; a few configurations could not be reached due to robot motion planner failure. In all cases, photographs of one or both icosahedrons in every pose were taken and added to the dataset. In addition, for every setup (self-touch, horizontal and vertical plane), 20 repetitions in 2 different positions were performed to evaluate the measurements' repeatability (Repeatability measurement dataset). For kinematic calibration, the distribution of robot joint angles is important. It is visualized in Fig. 6 for the different experiments.

**Laser tracker experiment** ($\boldsymbol{D}^{lt}$). We sampled the whole range of the first 4 joints of the manipulator and added uniform noise to the values to cover the whole range of joint angles. The joint angles of the last two joints were set so that the retroreflector faced the laser tracker. Configurations which would be in collision with the robot or the surrounding environment were excluded. This resulted in 685 configurations for the right arm, out of which 586 poses were actually recorded by the laser tracker (see Fig. 7; Fig. 8 for joint space distribution).

**Dataset structure.** The whole dataset $\boldsymbol{D}^{whole}$ contains $M$ dataset points (in fact, they are row vectors): $\boldsymbol{D}^{whole} = \{\boldsymbol{D}_1, \dots, \boldsymbol{D}_M\}$. Each dataset point $\boldsymbol{D}_j$ of the dataset $\boldsymbol{D}^{whole}$ consists of the assumed pose of right and left icosahedron centers $\boldsymbol{x}^{I,r}$ and $\boldsymbol{x}^{I,l}$, respectively (computed from forward kinematics); the joint configuration of the robot $\boldsymbol{\theta} = \{\theta_1^{ra}, \dots, \theta_N^{ra}, \theta_1^{la}, \dots, \theta_N^{la}\}$ ($ra$ and $la$ denoting right and left arm, respectively); coordinates of every marker ($K$ being the number of markers) in each of the cameras $\boldsymbol{u} = \{\boldsymbol{u}_1^{rc}, \dots, \boldsymbol{u}_K^{rc}, \boldsymbol{u}_1^{lc}, \dots, \boldsymbol{u}_K^{lc}\}$ ($rc$ and $lc$ denoting right camera and left camera, respectively); position of laser tracker ball retroreflector in the tracker coordinate system including uncertainties U95 [42] of position detection and the tracker measurement timestamp $\boldsymbol{L} = \{\mathbf{x}^L, u^L, t^L\}$. We also saved for reference the magnitude of force measured by both force sensors before ($F_b$) and during ($F_a$) the contact for each arm $\boldsymbol{F} = \{F_b^{ra}, F_a^{ra}, F_b^{la}, F_a^{la}\}$; and the names of saved camera images $n^r, n^l$. Individual dataset points are organized in a matrix, where each line $i$ of the matrix corresponds to the individual dataset point:

$$\boldsymbol{D}^{whole} = [\boldsymbol{D}_1, \dots, \boldsymbol{D}_M]^T, \text{where :} \\ \boldsymbol{D}_i = [\boldsymbol{x}_i^{I,r}, \boldsymbol{x}_i^{I,l}, \theta_i, \boldsymbol{F}_i, \boldsymbol{L}_i, n^r, n^l, \boldsymbol{u}_i]^T. \tag{4}$$

The whole dataset $\boldsymbol{D}^{whole}$ contains $M = 1268$ logged poses with 23 022 marker reprojections in total (12 371 from the right camera and 10 651 from the left camera), which makes approx. 25 marker reprojection per pose for self-contact experiment and 11 marker reprojections per pose for planar constraints. There are also 586 poses acquired by the laser tracker—this dataset does not contain marker reprojections. The reprojections are sorted from the lowest to the highest marker ID for every camera image. If a marker was not found in the image, its coordinates are denoted as $(NaN, NaN)$. Similarly, if data were not measured or captured by Leica tracker, all corresponding values are filled with $NaN$.

For optimization, we transformed the dataset so that one line would relate to one data point. That is, for a single robot configuration with multiple marker reprojections, the corresponding datapoint is unfolded into multiple rows—1 per marker detected. A number defining the robot configuration ($i$ from the original dataset $\boldsymbol{D}^{whole}$) is repeated on

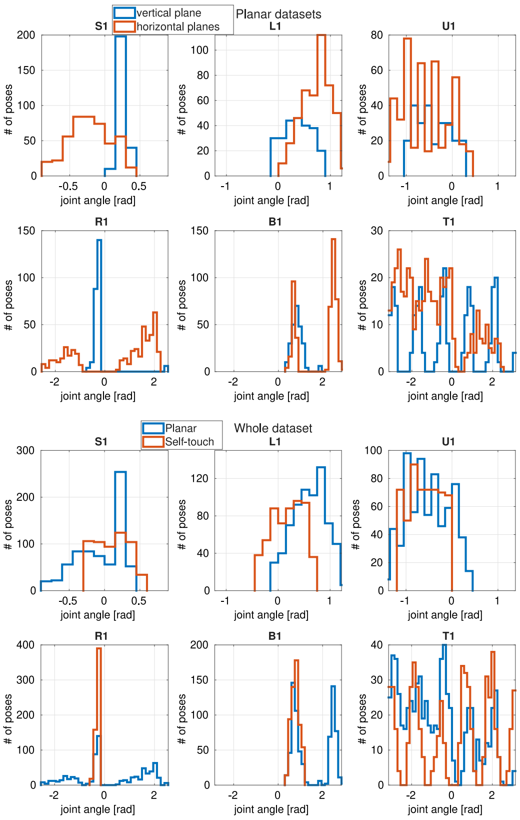


**Fig. 6.** Distributions of robot joint angles (S1,L1,U1,R1,B1,T1) for measured poses for the right arm across different experiments (only the right arm was evaluated in all experiments). "Planar datasets" subfigure compares the vertical plane contact in blue with the horizontal plane contact in red. "Whole dataset" subfigure compares the self-touch experiment in red with the combination of all planar setups (2 horizontal + 1 vertical plane) in blue.

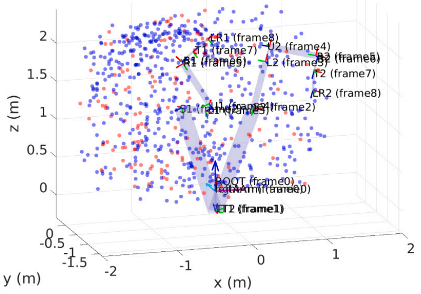


**Fig. 7.** Positions of the retroreflector in the Laser tracker dataset (right arm) with information whether the pose was additionally logged by the tracker or not (blue — captured by the tracker (586 poses), red — not captured by the tracker (99 poses) due to the poor visibility of the retroreflector). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

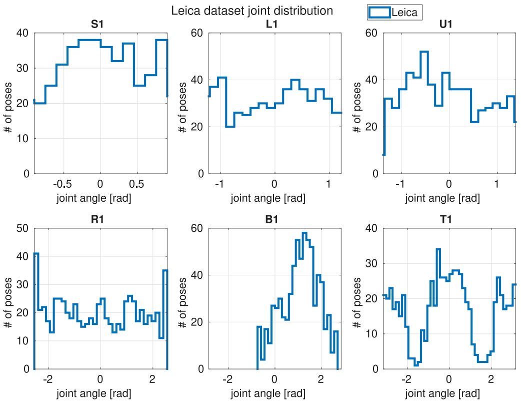


**Fig. 8.** Distribution of joint angles for Laser tracker dataset (right arm).

every line. Thus, one line consists of a number defining the robot pose, a face number of the detected marker, index of the arm (1 for right or 2 for left), index of the camera (1 for right or 2 for left), position of the marker center in the camera $\boldsymbol{u} = (u, v)$ in [px], and the current robot joint configuration (turntable, S1, L1, U1, R1, B1, T1, S2, L2, U2, R2, B2, T2) in [rad], position of the tracker ball retroreflector in the tracker coordinate system, and uncertainty U95 of the measurement.

The dataset and its description can be downloaded from [3]. For the positions including LEICA measurements, the appropriate csv file with $(x, y, z)$ positions detected by the laser tracker scanner are available.

## 4. Multi-chain robot calibration

In multi-chain robot calibration we estimate parameter vector $\boldsymbol{\phi} = \{[a_1, \dots, a_n], [d_1, \dots, d_n], [\alpha_1, \dots, \alpha_n], [o_1, \dots, o_n]\}$ with $k \in N$, where $N = \{1, \dots, n\}$ is a set of indices identifying individual links; $a_k$, $d_k$ and $\alpha_k$ are the first three parameters of the DH formulation of link $k$; $o_k$ is the offset that specifies the positioning of the encoders on the joints with respect to the DH representation. This is sometimes referred to as static geometric calibration. We often estimate a subset of these parameters only, assuming that the others are known. This subset can for example consist of a subset of links $N' \subset N$ (e.g., only parameters of one arm are to be calibrated) or a subset of the link parameters. Here we focus on offsets in the revolute joints $\mathbf{o}$ (sometimes dubbed "daily calibration" [47]).

Let $\boldsymbol{D} \subset \boldsymbol{D}^{whole}$ denote the set of robot configurations (dataset points) used for optimization:

$$\boldsymbol{D}_i = [m_i, c_i, \boldsymbol{u}_i, \boldsymbol{\theta}_i]$$

where $i \in \{1, \dots, M'\}$ is an index identifying one particular dataset point, $M'$ is the number of dataset points used for optimization, $m_i$ is face number of the detected marker, $c_i$ is the index of the used camera, $\boldsymbol{u}_i = (u_i, v_i)$ is the position of the marker center in the camera, and $\boldsymbol{\theta}_i$ is the current robot joint configuration (joint angles from joint encoders for the given robot configuration).

Estimation of the parameter vector $\boldsymbol{\phi}$ is done by optimizing a given objective function $\boldsymbol{f}(\boldsymbol{\phi}, \boldsymbol{D}, \boldsymbol{\zeta})$:

$$\boldsymbol{\phi}^* = \arg\min_{\boldsymbol{\phi}} \boldsymbol{f}(\boldsymbol{\phi}, \boldsymbol{D}, \boldsymbol{\zeta}), \tag{5}$$

$$\boldsymbol{f}(\boldsymbol{\phi}, \boldsymbol{D}, \boldsymbol{\zeta}) = \|g(\boldsymbol{\phi}, \boldsymbol{D}, \boldsymbol{\zeta})\|^2 = \sum_{i=1}^{M'} g(\boldsymbol{\phi}, \boldsymbol{D}_i, \boldsymbol{\zeta})^2, \tag{6}$$

where $M'$ is the number of robot configurations and corresponding end effector positions used for calibration (hereafter often referred to as

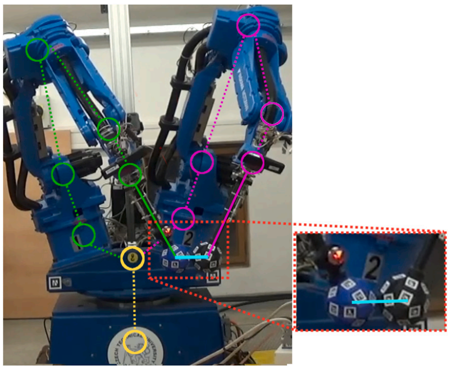


**Fig. 9.** Self-contact experiment. The left and right arm chains are drawn in purple and green, respectively. Cyan indicates the distance between end effector centers. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

"poses" for short), $\boldsymbol{\phi}$ is a given parameter estimate, dataset point $\boldsymbol{D}_i$ includes joint angles $\boldsymbol{\theta}_i$ from joint encoders for the given robot configuration, and constant vector $\boldsymbol{\zeta}$ defines all other necessary parameters such as camera calibration, fixed transformations, fixed DH parameters, or other properties of the robot. For chains involving cameras, the function $g(\boldsymbol{\phi}, \boldsymbol{D}_i, \boldsymbol{\zeta})$ refers to reprojection error as described in the next section while for the chains including contact, it corresponds to the distance between a real (observed) end effector position $\boldsymbol{p}_i^r$ and an estimated end effector position $\boldsymbol{p}_i^e$ computed using forward kinematic function for a given parameter estimate $\boldsymbol{\phi}$ and joint angles from joint encoders $\boldsymbol{\theta}_i$. For the case of planar constraints, the shortest distance of end effector position (icosahedron center) to the parameterized plane is minimized.

We study different combinations of intersecting chains and their performance in calibrating one another. Specific form of the function $g(\boldsymbol{\phi}, \boldsymbol{D}_i, \boldsymbol{\zeta})$ for individual considered chains and their combinations is discussed in the following subsections. At the end of every subsection, we also state how many components of the pose can be measured using the different methods—similarly to the analysis for standard open-loop and closed-loop calibration approaches in [1].

For problem representation, optimization, and some visualization, we employed the multisensorial calibration toolbox [48].

### 4.1. Self-contact – two arms chain (LA-RA)

This corresponds to the self-contact scenario in which contact occurs directly between the end effectors of the Left Arm (LA) and Right Arm (RA). As described in Section 3.1, for contact, the end effectors can be treated as spheres and the contact can occur between any of the 10 spherical tiles. The newly established kinematic chain for the upper body includes both arms; the head and eyes are excluded (see Fig. 9). To optimize parameters describing this chain, we minimize the distance between estimated positions in the 3D space of left and right arm end effectors (see Fig. 10).

In this case, the parameter vector $\boldsymbol{\phi}$ consists of the following parameters: $\boldsymbol{\phi} = \{\boldsymbol{\phi}^{ra}, \boldsymbol{\phi}^{la}\}$, where $\boldsymbol{\phi}^{ra}$ and $\boldsymbol{\phi}^{la}$ are DH parameters being calibrated corresponding to the robot right and left arm, respectively. The objective function to be optimized is:

$$\boldsymbol{g}^{st}(\boldsymbol{\phi}, \boldsymbol{D}^{st,u}, \zeta) = [c(\boldsymbol{\phi}, \boldsymbol{D}_1, \boldsymbol{\zeta}) - q(\boldsymbol{\zeta}), \dots, c(\boldsymbol{\phi}, \boldsymbol{D}_M, \boldsymbol{\zeta}) - q(\boldsymbol{\zeta})] \tag{7}$$

where the function $c(\boldsymbol{\phi}, \boldsymbol{D}_i, \boldsymbol{\zeta})$ computes the distance of the end effector centers in the configuration given by the dataset point $\boldsymbol{D}_i$, where $\boldsymbol{D}_i \in \boldsymbol{D}^{st,u}$, $\boldsymbol{D}^{st,u} \subset \boldsymbol{D}^{st}$ is a subset of the dataset points with a particular robot configuration (see Dataset structure in Section 3.6 for details). The distance of the end effector centers, marked as $q(\boldsymbol{\zeta})$, is equal to one icosahedron diameter $2r$, because both end effectors have identical shape. For the icosahedron diameter, we took the value from CAD model of $2r = 116$ mm and kept it fixed [46].

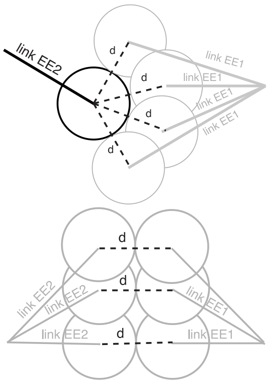


**Fig. 10.** Visualization of self-contact scenario. When end effectors are in contact, they are separated by distance $d$. This can happen in multiple configurations. For simplicity, we visualize the case when only end effector link length is estimated for one arm (left) or both arms (right), with the additional assumption that the link length is the same for both arms.

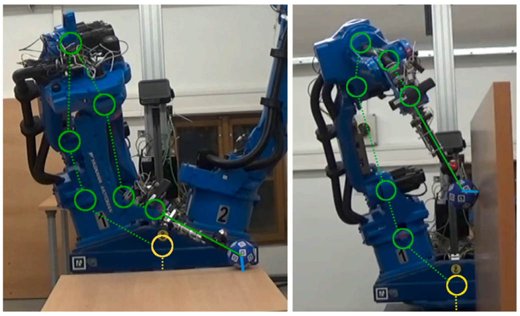


**Fig. 11.** Contact with planar constraints: horizontal plane (left), vertical plane (right).

As can be seen, the objective function contains a set of constraints on the distances between the end effector positions. These constraints can be written for the case of self-contact as follows: Let $(x_i^r, y_i^r, z_i^r)$ and $(x_i^l, y_i^l, z_i^l)$, be the center of right and left arm end effector computed from forward kinematics for data point $i$ with the given joint configuration $\boldsymbol{\theta}_i$, respectively. Then for each data point $\boldsymbol{D}_i$ in the dataset, we have the following constraint (left side of the equation corresponds to $c(\boldsymbol{\phi}, \boldsymbol{D}_i, \boldsymbol{\zeta})$ and right side to $q(\boldsymbol{\zeta})$ in Eq. (7)):

$$\sqrt{(x_i^l - x_i^r)^2 + (y_i^l - y_i^r)^2 + (z_i^l - z_i^r)^2} = 2r \tag{8}$$

According to [1], this corresponds to restricting 1 position parameter. The parameters of the touched surface ($2r$) are in this case known—same as for planar constraints with known plane parameters.

### 4.2. Planar constraints — one arm chain in contact with a plane (LA/RA)

This corresponds to the scenario where the Left (LA) or Right (RA) robotic arm is getting into contact with a plane (see Fig. 11). In this type

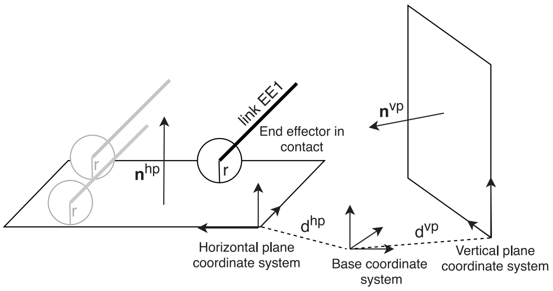


**Fig. 12.** Visualization of planes with their corresponding parameters ($\boldsymbol{n}^{hp}$, $d^{hp}$ and $\boldsymbol{n}^{vp}$, $d^{vp}$) and coordinate frames. End effector with radius $r$ in contact with horizontal plane in multiple places is shown.

of optimization problem, we can distinguish formulations including single-plane or multiple-plane constraints [24,25]. The classical formulations of the problem use either a general equation of the constraint plane or plane normals [23]. The general equation of a plane is:

$$ax + by + cz + d = 0, \tag{9}$$

where $\mathbf{n} = (a, b, c)$ is a plane normal vector. The parameters of the plane can be known in advance (as in [23,26] or [20] where calibration cube is used), or unknown (as in our case or [24]). The planes and their corresponding parameters are visualized in Fig. 12.

When the parameters of the plane are unknown, the parameter vector $\boldsymbol{\phi}$ consists of the following parameters: $\boldsymbol{\phi} = \{\boldsymbol{\phi}^{ra/la}, \mathbf{n}, d\}$, where $\boldsymbol{\phi}^{ra/la}$ are the DH parameters of the robot arm in contact, $\mathbf{n}$ is a plane normal, and $d$ is the distance of the plane from the origin. We formulated the objective function as the distances between contacts and a single or multiple fitted planes:

$$g^p(\boldsymbol{\phi}^p, \boldsymbol{D}^p, \boldsymbol{\zeta}) = \Big[\mathbf{c}(\boldsymbol{\phi}^{hp1}, \boldsymbol{D}^{hp1}, \boldsymbol{\zeta}) - r, \; \mathbf{c}(\boldsymbol{\phi}^{hp2}, \boldsymbol{D}^{hp2}, \boldsymbol{\zeta}) - r, \mathbf{c}(\boldsymbol{\phi}^{vp}, \boldsymbol{D}^{vp}, \boldsymbol{\zeta}) - r\Big], \tag{10}$$

where $\boldsymbol{D}^p \subset \boldsymbol{D}^{whole}$, $\boldsymbol{D}^p = \{\boldsymbol{D}^{hp1}, \boldsymbol{D}^{hp2}, \boldsymbol{D}^{vp}\}$ is a set of datasets where contacts between the end effector and lower horizontal plane ($\boldsymbol{D}^{hp1}$), higher horizontal plane ($\boldsymbol{D}^{hp2}$) or vertical plane ($\boldsymbol{D}^{vp}$) were performed. The set $\boldsymbol{\phi}^p = \{\boldsymbol{\phi}^{hp1}, \boldsymbol{\phi}^{hp2}, \boldsymbol{\phi}^{vp}\}$ is a set of parameters for lower horizontal plane ($\boldsymbol{\phi}^{hp1}$), higher horizontal plane ($\boldsymbol{\phi}^{hp2}$), and vertical plane ($\boldsymbol{\phi}^{vp}$), respectively. The vector $\mathbf{c}(\boldsymbol{\phi}^j, \boldsymbol{D}^j, \boldsymbol{\zeta})$ is a vector of distances between individual end effector positions and the given plane j for each datapoint $\boldsymbol{D}^j_i$ from the given dataset $\boldsymbol{D}^j$. The distance is computed using plane normals and corresponding plane coordinates as follows: $c(\boldsymbol{\phi}^j, \boldsymbol{D}^j_i, \boldsymbol{\zeta}) = \|\boldsymbol{n}^j \boldsymbol{p}^j_i(\boldsymbol{\phi}^{j,ra/la}) + d\|$. Point $\mathbf{p}^j_i$ is the center of the end effector computed by forward kinematics from dataset point $\boldsymbol{D}^j_i$; $\boldsymbol{\phi}^{j,ra/la}$ is the estimated parameter vector corresponding to the DH parameters of the touching arm; $\mathbf{n}^j = [a\ b\ c]$ is the plane normal; $d^j$ is the distance of the plane from the origin. These plane parameters ($\mathbf{n}^j$ and $d^j$) are estimated at each iteration of the optimization process based on current point coordinates estimates by SVD method as described below. The $\boldsymbol{\zeta}$ wraps up all other necessary parameters.

To acquire parameters of the fitted plane in each iteration, the measured points are converted to homogeneous coordinates and their center of gravity is computed and subtracted from all points. Afterwards, Matlab function *SVD* is called. The singular vector corresponding to the smallest singular value is set as a normal of the plane. Parameter $d$ in Eq. (9) is calculated from:

$$d = -ax_0 - by_0 - cz_0, \tag{11}$$

where $(x_0, y_0, z_0)$ are coordinates of the points center.

Let $\boldsymbol{n} = [a, b, c]$ be a plane normal, $d$ the distance of the plane from the origin, $(x^r_i, y^r_i, z^r_i)$ the center of the right arm end effector computed from forward kinematics for data point $\boldsymbol{D}_i$ with the given joint configuration $\theta_i$, and $r$ the radius of the icosahedron. Then for each data point $\boldsymbol{D}_i$ in the dataset $\boldsymbol{D}^{hp/vp}$ we get the following constraint (left side of the equation corresponds to $\mathbf{c}(\boldsymbol{\phi}^{vp/hp}, \boldsymbol{D}^{vp/hp}, \boldsymbol{\zeta})$ in Eq. (10)):

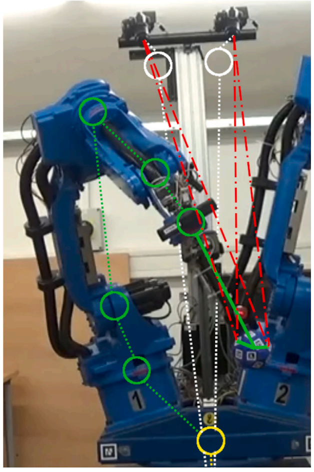


**Fig. 13.** Self-observation chains: Red denotes reprojection from individual AruCo markers to left and right camera.

$$\sqrt{(ax^r_i + by^r_i + cz^r_i + d)^2} = r \tag{12}$$

As per [1], this type of calibration corresponds to the restriction of 1 position parameter. Compared to the self-contact setup, in the case of planes with unknown parameters (our case), new calibration parameters (parameters of the plane $\boldsymbol{n}$ and $d$) have to be added.

### 4.3. Self-observation by cameras (LA-LEye, LA-REye, RA-LEye, RA-REye)

This corresponds to the scenario where we observe the Left Arm (LA) or Right Arm (RA) end effector with AruCo markers via Left Camera (LEye) or Right Camera (REye) (see Fig. 13). We calibrate: (i) extrinsic parameters of the cameras (in our case as DH links) while assuming the robot DH parameters to be known, or (ii) the whole kinematic chain of the robot arm simultaneously with camera extrinsic parameters. In this case, the optimization is done by minimizing the reprojection error between the observed AruCo markers' positions in the camera and the estimated position using the current estimated kinematic model.

To obtain projected marker positions (which determine end effector position) in each of the robot cameras, we apply the calibrated camera model. The camera was precalibrated using a standard camera intrinsic calibration (see Section 3.5). The extrinsic parameters (in our case expressed in DH parameters) of the camera were precalibrated based on reprojections of ArUco markers using the fixed nominal parameters of the arm. First, we have to transform marker positions to the camera frame:

$$\begin{bmatrix} x_c & y_c & z_c & 1 \end{bmatrix}^T = \boldsymbol{T}_m^{camera} \cdot \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T,$$

where $[x_c, y_c, z_c, 1]^T$ are homogeneous coordinates of marker position in the frame of the given camera and $\boldsymbol{T}_m^{camera}$ is a transformation from the marker $M_m$ to the given camera achieved through standard approach for forward kinematics using DH parameters. Afterwards, we apply a standard pinhole camera model extended with radial and tangential distortion coefficients and transform the 3D point in camera frame ($[x_c, y_c, z_c]$) into image coordinates $[u, v]$ (2D plane of the camera) (see camera model in Section 3.5). The actual marker position means the center of the ArUco marker. The OpenCV function *calibrateCamera* provides calibration with resolution of whole pixels. ArUco marker detection is done by the OpenCV function *cv2.aruco.detectMarkers*, which provides the coordinates of all four marker corners. From these, the center of the marker is calculated as an intersection of the diagonals connecting the marker corners. The error resulting from this assumption is smaller than the calibration error [46].

The parameter vector $\boldsymbol{\phi}$ consists of the following parameters: $\boldsymbol{\phi} = \{\boldsymbol{\phi}^{ra/la}, \boldsymbol{\phi}^{rc/lc}\}$, where $\boldsymbol{\phi}^{ra}$, $\boldsymbol{\phi}^{la}$, $\boldsymbol{\phi}^{rc}$, and $\boldsymbol{\phi}^{lc}$ are calibrated DH parameters corresponding to the right arm, left arm, right camera, and left camera, respectively. The objective function is formulated as the distance between projected markers and their pixel coordinates in the images:

$$\begin{aligned} \boldsymbol{g}^{so}(\boldsymbol{\phi}, \boldsymbol{D}^{so}, \boldsymbol{\zeta}) = \big[&p(\boldsymbol{\phi}, \boldsymbol{D}_1, \boldsymbol{\zeta}) - z(\boldsymbol{D}_1), \dots, \\ &p(\boldsymbol{\phi}, \boldsymbol{D}'_M, \zeta) - z(\boldsymbol{D}'_M)\big] \end{aligned} \tag{13}$$

where $p(\boldsymbol{\phi}, \boldsymbol{D}_i, \boldsymbol{\zeta})$ is the reprojection of marker $m_i$ from dataset point $\boldsymbol{D}_i$ where $i \in \{1, \dots, M'\}$, $M'$ is the length of the dataset $\boldsymbol{D}$. The $z(\boldsymbol{D}_i)$ is the actual marker position in the camera image.

This approach does not require information from both eyes and enables us to estimate only one side of the robot body (e.g., parameters of the left arm and left camera). For example, the estimated parameter vector $\boldsymbol{\phi}$ in the case of the kinematic chain connecting left arm and left camera consists of the following parameters: $\boldsymbol{\phi} = \{\boldsymbol{\phi}^{l}, \boldsymbol{\phi}^{lc}\}$, where $\boldsymbol{\phi}^{l}$ and $\boldsymbol{\phi}^{lc}$ are parameters corresponding to the robot left arm and to the left camera, respectively.

The conditions in the objective function can be expressed as follows: Let $\boldsymbol{x}^m$ be a center of origin of marker $m_i$ (obtained by forward kinematics for the given DH parameters estimate), $\boldsymbol{x}^c$ be a marker position in the given camera frame, $\boldsymbol{T}_{m_i}^{EEF}$ be a transformation from marker $m_i$ to the arm end effector (transforms to individual ArUco markers can be found at the dataset page at [3]) and $\boldsymbol{T}_{EEF}^{camera}$ transformation from the end effector to the given camera frame, $p(\boldsymbol{x}^c)$ a reprojection of 3D point in camera frame to image coordinates $[u, v]$ (see Eq. (1)):

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = p(\boldsymbol{x}^c) = p(\boldsymbol{T}_{EEF}^{camera} \boldsymbol{T}_{m}^{EEF} \cdot \boldsymbol{x}^m).$$

Then, for each data point $i$ (corresponding to position of marker $m_i$ in the configuration $\boldsymbol{\theta}_i$), reprojection of marker $m_i$ to camera frame $[u_i, v_i]$, and each measured marker position in the camera image $(u_i^m, v_i^m)$ we get two equations:

$$\begin{aligned} u_i &= u_i^m, \\ v_i &= v_i^m. \end{aligned} \tag{14}$$

According to [1], this type of calibration corresponds to the restriction of 2 position parameters. Still, we have to add camera DH parameters (see Table 2) to parameters being calibrated to enable the reprojection of markers to the camera frame.

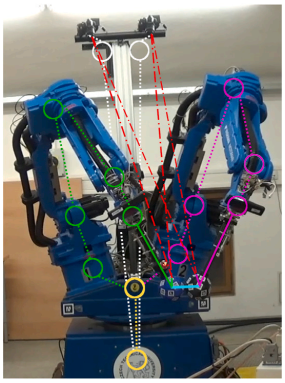


Fig. 14. Illustration of calibration combining multiple chains: self-contact and self-observation. All chains originate in a common base frame (bottom yellow circle). The left and right arm chains are drawn in purple and green, respectively. The eye chains are drawn in white. Red lines denote reprojection into the cameras. The cyan mark indicates the distance between end effector centers (one diameter). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

### 4.4. Combining multiple chains (LA-RA-LEye, LA-RA-LEye-REye)

In order to estimate all kinematic parameters of the robot, we can take advantage of combining some or all of the above-mentioned kinematic chains. For example, in the case that we combine LA-RA, LA-LEye and LA-REye chains together into LA-RA-LReye (see Fig. 14), the estimated parameter vector $\boldsymbol{\phi}$ consists of the following parameters: $\boldsymbol{\phi} = \{\boldsymbol{\phi}^{r}, \boldsymbol{\phi}^{l}, \boldsymbol{\phi}^{rc}, \boldsymbol{\phi}^{lc}\}$, where $\boldsymbol{\phi}^{l}$, $\boldsymbol{\phi}^{r}$, $\boldsymbol{\phi}^{rc}$, and $\boldsymbol{\phi}^{lc}$ are parameters corresponding to the left arm, right arm, right camera, and left camera, respectively. Similarly, contact of right arm with a horizontal and vertical plane can be combined with self-observation by right camera, resulting in the parameter vector $\boldsymbol{\phi}$: $\boldsymbol{\phi} = \{\boldsymbol{\phi}^{r}, \boldsymbol{\phi}^{rc}, \boldsymbol{n}^{hp}, d^{hp}, \boldsymbol{n}^{vp}, d^{vp}\}$, where $\boldsymbol{n}^{hp}$ and $d^{hp}$ are parameters defining the horizontal plane, and $\boldsymbol{n}^{vp}$ and $d^{vp}$ are parameters defining the vertical plane.

The overall objective function can be generally defined as (depending on which datasets and criteria we want to use for calibration):

$$\begin{aligned} \boldsymbol{g}(\boldsymbol{\phi}, D, \zeta) = \big[&\boldsymbol{k}^{st} \odot \boldsymbol{g}^{st}(\boldsymbol{\phi}, \boldsymbol{D}^{st}, \zeta), \boldsymbol{k}^{p} \odot \boldsymbol{g}^{p}(\boldsymbol{\phi}, \boldsymbol{D}^{p}, \zeta), \\ &\boldsymbol{k}^{so} \odot \boldsymbol{g}^{so}(\boldsymbol{\phi}, \boldsymbol{D}^{so}, \zeta)\big], \end{aligned} \tag{15}$$

where $\boldsymbol{D}^{st}$, $\boldsymbol{D}^{p} = \{\boldsymbol{D}^{hp1}, \boldsymbol{D}^{hp2}, \boldsymbol{D}^{vp}\}$ and $\boldsymbol{D}^{so} = \{\boldsymbol{D}^{st}, \boldsymbol{D}^{p}\}$ are datasets for self-touch, planar constraints optimization, and self-observation, respectively. Parameters $\boldsymbol{k}^{st}, \boldsymbol{k}^{p}$, $\boldsymbol{k}^{so}$ are scale factors to reflect the different uncertainty/reliability of the components, the number of measurements per configuration, and transformations from distance errors given in meters with the reprojection errors in pixels. Symbol $\odot$ marks a Hadamard product: i.e. $(\boldsymbol{k}^{st} \odot \boldsymbol{g}^{st})_i = k_i^{st} \cdot g_i^{st}$. The value of these parameters is set independently for each pose: $k_i^{st} = \eta_i^{st} \cdot p^{st} \cdot \mu_i$, $k_i^{p} = \eta_i^{p} \cdot^{p} \cdot \mu_i$, and $k_i^{so} = \eta_i^{so} \cdot p^{so}$, where $\eta^{st}, \eta^{p}$, $\eta^{so}$ reflect the reliability of the measurement (e.g., $\eta_i = \sigma_i^{-\frac{1}{2}}$, where $\sigma_i$ is the uncertainty of the measurement in the given pose). In this work, $\eta = 1$ was used for all approaches. The parameter $p_i$ reflects the fact that there are multiple markers detected by cameras for the given contact configuration. Therefore, in the case of (planar constraints/self-touch) contact and self-observation combination $p^p = 10$, $p^{st} = 20$ (there are two icosahedrons in contact and on average 20 marker detections per contact event), and $p^{so} = 1$. The coefficient $\mu_i$ is determined from intrinsic parameters of cameras (60 deg horizontal view angle, image size 4000 × 6000px) and distance $d_i$ of the end effector from the camera: $\mu_{i,x} = 4000\text{px}/(d_i(\pi/3))$, $\mu_{i,y} = 6000\text{px}/(d_i(\pi/3))$. For simplicity,

$\mu_{i,x} = \mu_{i,y}$ was used. Fig. 14 shows connections of different calibration chains and constraints (e.g., distance between end effectors during a self-contact or distance between end effector and a plane for a plane constraint).

### 4.5. Calibration using laser tracker

In this scenario, the robot arm with attached retroreflector is moving in free space to the configurations selected by sampling a joint space in the way that the retroreflector is facing the laser tracker (more details under Laser tracker experiment in Section 3.6). The distance between the position of the retroreflector acquired by the laser tracker and the position computed from current robot arm DH parameters (plus current joint angle values and using forward kinematics) is minimized.

The parameter vector $\boldsymbol{\phi}$ consists of the following parameters: $\boldsymbol{\phi} = \{\boldsymbol{\phi}^{ra/la}, \boldsymbol{R}, \boldsymbol{T}\}$, where $\boldsymbol{\phi}^{ra/la}$ are DH parameters corresponding to the right/left robot arm (with the link EEL1/EEL2 to the retroreflector — see Table 1), $\boldsymbol{R}$ and $\boldsymbol{T}$ are rotation and translation matrices defining laser tracker position w.r.t. the robot base frame.

The objective function is formulated as the error of distances:

$$g^L(\boldsymbol{\phi}, D, \boldsymbol{\zeta}) = \left[p(\boldsymbol{\phi}, \boldsymbol{D}_1, \boldsymbol{\zeta}), \ldots, \; p(\boldsymbol{\phi}, \boldsymbol{D}_M, \boldsymbol{\zeta})\right], \tag{16}$$

where the function $p(\boldsymbol{\phi}, \boldsymbol{D}_i, \boldsymbol{\zeta}) = \|\boldsymbol{x}_i^L - \boldsymbol{x}_i^r\|$ computes the distance of the transformed point $\mathbf{x}_i^L$ from laser tracker and the point $\mathbf{x}_i^r$ gained from forward kinematics and current estimate of robot DH parameters in the configuration given by the dataset point $\boldsymbol{D}_i$.

To calibrate the robot DH parameters, we minimize the distance between these 2 sets of 3D points (set $\mathbf{X}^L$ contains points from laser tracker in its coordinate system and set $\mathbf{X}^r$ includes points in the base coordinate system computed from a joint configuration using forward kinematics and the current estimate of robot DH parameters) using an iterative approach.

In each iteration of the optimization process we:

1. recompute the estimate of robot DH parameters
2. recompute rotation and translation matrix defining laser tracker position w.r.t. base frame. The relation between corresponding points in sets is generally:

   $$\mathbf{x}_i^r = \boldsymbol{R}\mathbf{x}_i^L + \boldsymbol{T} + N_i, \tag{17}$$

   where $\boldsymbol{R}$ and $\boldsymbol{T}$ are rotation and translation matrices defining laser tracker position w.r.t. the robot base frame, $N_i$ is noise for the $i$th datapoint. We used an algorithm introduced by Arun et al. [49] for finding least-squares solution of $\boldsymbol{R}$ and $\boldsymbol{T}$. It is a non-iterative algorithm using the singular value decomposition of a 3 × 3 matrix.

This is a standard open-loop calibration method in which 3 components of the pose are measured: the 3D position, not orientation, is acquired from the laser tracking system [1]. The transform ($\boldsymbol{R}$, $\boldsymbol{T}$) relating the robot base frame to the laser tracker frame of reference has to be added to calibration. Since we express rotation by rotation vector, this corresponds to adding 6 parameters to calibration.

### 4.6. Non-linear least squares optimization

For solving the optimization problem, the Levenberg–Marquardt iterative algorithm was employed. This is a standard choice for kinematic calibration (e.g., [2,14,50]) and combines the advantages of gradient descent and Gauss–Newton algorithms, which can be also applied to this problem. For implementation, we used the Matlab Optimization Toolbox and the nonlinear least-squares solver *lsqnonlin* with the Levenberg–Marquardt option and parameters *typicalx* and *scaleproblem*.

### 4.7. Observability and identifiability of parameters

According to [51], the observability index measures the quality of the dataset based on the identification Jacobian matrix $\mathbf{J}$, which represents the sensitivity of minimized values to the change of individual parameters. Borm and Menq [52] proposed a measure $O_1$; Driels and Pathre [53] proposed $O_2$; Nahvi and Hollerbach proposed measures $O_3$ [54] and $O_4$ [55]. All these measures can be computed from the singular value decomposition of $\mathbf{J}$. They are defined as:

$$O_1 = \frac{(\sigma_1\sigma_2\ldots\sigma_m)^{1/m}}{\sqrt{n}}, \quad O_2 = \frac{\sigma_{min}}{\sigma_{max}},$$
$$O_3 = \sigma_{min}, \quad O_4 = \frac{\sigma_{min}^2}{\sigma_{max}}, \tag{18}$$

where $\sigma_j$ is $j$th singular number, $m$ is the number of independent parameters to be identified and $n$ is the number of calibration configurations.

The identification Jacobian matrix itself shows us the identifiability of individual optimized parameters: $\mathbf{J}(i,j) = \frac{\partial X_i}{\partial \phi_j}$, where $X_i$ is a distance (Eqs. (7) and (10)) or a reprojection error (Eq. (13)) and $\phi_j$ is the parameter to be estimated. If a matrix column related to a parameter consists only of zeros, the parameter is unidentifiable which leads to an unobservable problem (the minimal singular number is zero). According to [1], an unidentifiable parameter means that the experimental setup does not allow it to be identified, not that it is intrinsically unidentifiable. The identifiability can be improved by adding additional sensors to the setup as well as by extending the amount of poses in the dataset. In our analysis, we compare observability indices $O_1$ (representing the volume of a hyperellipsoid specified by singular numbers) and $O_4$ (noise amplification index which measures both eccentricity of the hyperellipsoid through $O_2$ and size of the hyperellipsoid through $O_3$) (see [1] for an overview) for individual chains and estimated parameter vectors.

### 4.8. Perturbation of the initial parameters estimate

To evaluate the dependence of the optimization performance on the quality of the initial estimates of the parameters, we perturbed all estimated parameters by a *perturbation factor* $p = \{1, 3, 10\}$ (in experimental section, we show results for $p = 3$). We perturbed all initial offset values $o_i$ as follows:

$$o_i^{new} = 0.1p \cdot U(-1,1) + o_i \; [rad], \tag{19}$$

where $U(a,b)$ is uniform distribution between $a$ and $b$. It is reasonable to expect that the remaining DH parameters ($\alpha$, $a$, and $d$) will be in general more accurate as they can be extracted from CAD models and there is no moving part and no encoder involved. Therefore, their perturbation was chosen as follows:

$$\begin{aligned} \alpha &: \alpha_i^{new} = 0.01p \cdot U(-1,1) + \alpha_i \; [rad], \\ a, d &: \Phi_i^{new} = 0.01p \cdot U(-1,1) + \Phi_i \; [m]. \end{aligned} \tag{20}$$

### 4.9. Evaluation

Before optimization, we randomly divided the dataset into training and testing data with a ratio of approximately 70 to 30. Training and testing datasets contained distinct sets of poses—multiple observations of markers in a given pose were all added to the same batch. Optimization was based on the training data; testing data were used to evaluate the result. We used root-mean-square (RMS) error to compare the results from multiple optimization runs. It is computed as:

$$RMS_c = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(k_i^c g^c(\boldsymbol{\phi}, \boldsymbol{D}_i, \boldsymbol{\zeta}))^2} =$$
$$\sqrt{\frac{1}{M}\|\boldsymbol{k}^c \odot \boldsymbol{g}^c(\boldsymbol{\phi}, \boldsymbol{D}^c, \boldsymbol{\zeta})\|^2}, \tag{21}$$

where $M$ is the number of observations/measurements, $k^c$ is a scale factor for the given calibration approach $c$ (see Section 4.4) and $g^c(\boldsymbol{\phi}, \boldsymbol{D}_i^c, \boldsymbol{\zeta})$ is the corresponding objective function (see Sections 4.1, 4.2, 4.3 or 4.5 for specific form of individual objective functions). Symbol $\odot$ marks a Hadamard product: i.e. $(\boldsymbol{k}^{st} \odot \boldsymbol{g}^{st})_i = k_i^{st} \cdot g_i^{st}$.

### 4.10. Accuracy of measuring individual components

The accuracy of several components constitutes a lower bound on the overall accuracy. In particular, we estimated or experimentally evaluated the accuracy of the following (*we note in brackets for which chains is the accuracy of the given component relevant*):

- **3D printed parts and their dimensions**: 0.1 mm error based on the printer specification; we assume the end effector rod to be straight. (*self-contact (Section 4.1) and contact with a plane chains (Section 4.2)*)
- **Intrinsic camera calibration**: we evaluated error of intrinsic camera calibration using multiple calibration patterns split to training and testing dataset. The resulting error is 0.73 mean reprojection error for calibration points in pixels on the testing set. This error is composed of the accuracy of detecting the dot pattern and the calibration itself. (*self-observation chains (Section 4.3)*)
- **Pose extraction from ArUco markers**: The calibration object with the Aruco markers was mounted on a linear positioning table. The object was moved along straight lines with a 0.01 mm precision. We captured images of the object in different positions and detected centers of the observed Aruco markers as shown in Fig. 15. Due to camera reprojection, the markers move along lines in the undistorted image (radial undistortion). We estimate the error of marker detection as distance of the detected markers from interpolated straight line (Fig. 15). The mean error is around 0.33 pixels for well visible markers (ID 107, 112, 113) and 0.63 pixels for other markers with maximal error of 1.5 pixels. We investigate the error along the line with similar results. Intrinsic camera calibration puts a lower bound on these results. (*self-observation chains (Section 4.3)*)
- **Repeatability of measurements**: The details of robot movements and how they were stopped once contact was detected are described in Section 3.3. We used the laser tracker to measure the repeatability of these movements, using 20 repetitions of the same movement at two different positions sampled from the grid for planar constraints (Fig. 5). These involved only small robot movements (10 cm) and repeatability was high – Fig. 16. For self-touch, larger movements from the robot home position were executed. However, self-contact experiments involved both large movements and small movements similar to those for contact with a plane. Thus, the statistics for Fig. 16 was combined for the self-touch distribution, resulting in overall lower repeatability. Results in $x$-coordinate are shown. Similar distribution and range of errors was observed for $y$ and $z$ coordinate. *(all chains)*
- **Camera projection error propagation**: Cameras measure projections to the image plane, producing higher uncertainty in the $z$-axis of the image coordinate system (image depth). The uncertainties in all three axes can be obtained by error propagation from object position uncertainty. The projection uncertainty can be written as a quadric equation $u^T Q u = 1$, where $Q$ is a matrix ($4 \times 4$ identity matrix assumed here) and $u$ is a $4 \times 1$ vector of both cameras' projection pixel coordinates. The projections vector $u$ can be expressed as $u = J_x X$, where $J_x$ is the Jacobian matrix computed from projection equations (Eq. (1)) and $X$ are the projected point coordinates. Combining these two equations, we obtain the equation for position uncertainty $X^T Q_x X = 1$, where $Q_x = J_x^T Q J_x$, which can be interpreted as an ellipsoid. Then, the eigenvalues of $Q_x$ are $a^{-2}$, $b^{-2}$, $c^{-2}$, where $a$, $b$, $c$ are half the principal axes' length in meters. In our case, the mean values are $a = 0.146$ mm, $b = 0.150$ mm, $c = 1.330$ mm, i.e. an error of 1 pixel corresponds to more than 1 mm error in object position in one axis. The comparison of uncertainties can be seen in Fig. 16. The length of the last axis of the ellipsoid (image depth) is one order of magnitude higher than the other lengths and the touch uncertainty (*self-observation chains (Section 4.3)*).

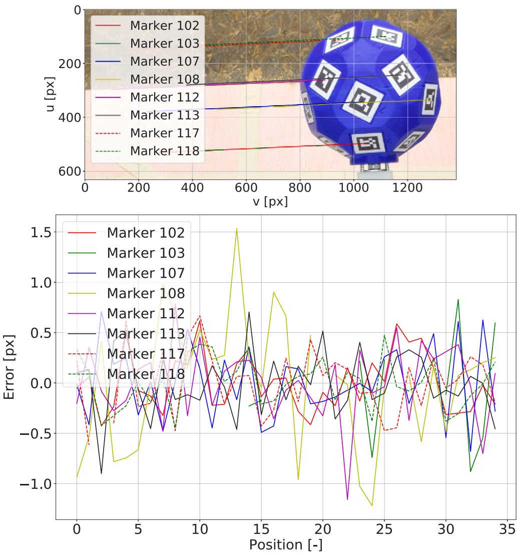


**Fig. 15.** Pose extraction from ArUco markers. (top) Icosahedron moving along a straight line with markers positions reprojected to camera frame ($u$ and $v$ are coordinates of the camera image). (bottom) Errors measured as perpendicular distance from the line for individual markers at specific positions.

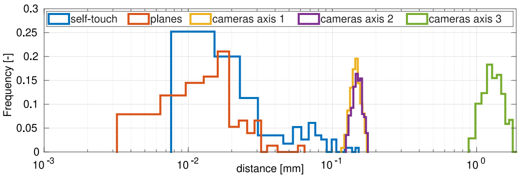


**Fig. 16.** Error distribution of different components. Data for self-touch and planes obtained from repeatability experiments. Camera errors from camera resolution and 3D space projection from robot workspace.

Based on the above-mentioned analysis, we can see that the lower bound of error for self-observation will consist of the combination of intrinsic camera calibration error ($e^i$) and ArUco marker detection error ($e^{ArUco}$): $e^{so} = e^i \bigoplus e^{ArUco}$. The measured displacement of the end effector between contacts is influenced by $e^{ArUco}$ and the observed error is systematic. In the worst case, we can estimate the error to be independent on the direction of the impact and for all directions consider the maximum observed error. As can be seen, repeatability measurements show that the accuracy is very high and the errors are below 0.03 mm. Finally, an analysis of camera errors in 3D space reveals their limited accuracy, in particular in one axis corresponding to the depth.

## 5. Experimental results

To evaluate and compare the results of kinematic parameters calibration across individual approaches and their combinations, we show results for the right arm of the robot only—this kinematic chain can be calibrated using all the datasets collected. First, we show the results for end effector length and angle offset (Section 5.1) without and with camera calibration. Afterward, we show results of "daily calibration" (calibrating only joint angle offsets $o$) of the whole right robot arm (including length of the end effector), consisting of: reference values acquired by the laser tracker (Section 5.2), calibration by individual approaches without and with camera calibration (Section 5.3). Finally, the results for all DH calibration are shown in Section 5.6 and evaluated on the independent laser tracker dataset.

The following notation/labeling of experiments is used for the individual evaluated approaches: *self-contact/self-touch* (Section 4.1); *1 horizontal plane, 2 horizontal planes, all planes* (2 horizontal and vertical plane) (Section 4.2); *all* (combination of all planes and self-contact calibration, Section 4.4); *self-observation* (Section 4.3).

Each dataset (see Section 3.6 for details) was split into training and testing part (70:30). The resulting RMS errors (according to Eq. (21)) for both training and testing datasets are shown and compared to the case where nominal parameters are used (*before*). We show resulting RMSE separately for self-contact distances between icosahedrons (*'dist'*), distances to plane (*'plane dist'*), and for camera reprojections in pixels (*'mark.'*) (e.g., Fig. 17). In addition, we investigate corrections to the parameter values and their mean and variance.

Calibration methods relying on physical contact—self-contact and planar constraints—can be employed independently or in combination with calibration using cameras (self-observation). We distinguish three possibilities that will be used throughout the rest of the Experimental results section:

- **No cameras.** Only contact-based calibration was employed and cameras were not used at all.
- **Fixed cameras.** Cameras' DH parameters were precalibrated (see Table 2) using the contact-based datasets (self-contact and contact with plane) from marker reprojection errors, using nominal values of the robot kinematic parameters. Then, to calibrate the robot right arm, reprojection of markers to the camera frame is used together with contact information, while the camera extrinsics stay fixed.
- **Camera position calibration (Cam. calib.).** Cameras were precalibrated in the same way as in *Fixed cameras*. During robot kinematic calibration, reprojection errors are combined with contact-based constraints. Additionally, camera extrinsics (expressed in DH parameters) are subject to calibration as well.

*Self-observation (S-O)*, i.e., information from camera reprojection errors, can also be employed independently. The datasets used are those featuring contact ($D^{st}$, $D^{hp}$, $D^{whole}$, etc.), but the constraints arising from physical contact are not employed.

Table 4 provides an overview which parameters can be subject to calibration under the different approaches. The parameters that can be calibrated using the contact-based approaches are marked in black. Fixed cameras enable calibration of the end effector orientation ($o_{EE1}$, in green; contact information alone does not provide any information about orientation — see Figs. 10 and 12). Camera extrinsic parameters, which may be also subject to calibration, are shown in blue.

### 5.1. Calibration of end effector length and joint angle offset

First, we evaluated the ability to calibrate the length and joint angle offset of the last link (EE1/EE2) by individual approaches. The diameter of the final part of the custom end effector—assuming it is a sphere in this case as the contact is at the spherical tiles—is known. The parameter being calibrated in all cases is the end effector length $d_{EE1}$; with fixed cameras, the orientation, $o_{EE1}$ can be also calibrated. In the case of planar constraints, the plane parameters $\{\boldsymbol{n}^{hp/vp}, d^{hp/vp}\}$ are estimated in addition.

**Table 4**
Overview of calibrated parameters. Parameters in green are added to the corresponding approach under *fixedCameras*. When using *calibCameras*, all the blue parameters are added. See Tables 1 and 2 for details about DH parameters.

| Calibration | Calibrated parameters |
|---|---|
| End effector | $d_{EE1}$, $o_{EE1}$ |
| Offsets | $o_{L1}$, $o_{U1}$, $o_{R1}$, $o_{B1}$, $d_{EE1}$, $o_{EE1}$ |
| Offsets by ext. device | $o_{L1}$, $o_{U1}$, $o_{R1}$, $o_{B1}$, $a_{EEL}$, $d_{EEL}$, $o_{EEL}$ |
| All DH parameters | $a_{L_1}$, $d_{L_1}$, $o_{L1}$, $a_{U_1}$, $d_{U_1}$, $\alpha_{U_1}$, $o_{U1}$, $a_{R_1}$, $d_{R_1}$, $\alpha_{R_1}$, $o_{R1}$, $a_{B_1}$, $d_{B_1}$, $\alpha_{B_1}$, $o_{B1}$, $a_{T1}$, $\alpha_{T1}$, $d_{EE1}$, $o_{EE1}$ |
| Camera calibration | $a_{TT3}$, $d_{TT3}$, $\alpha_{TT3}$, $o_{TT3}$, $d_{C1}$, $o_{C1}$, $a_{TT4}$, $d_{TT4}$, $\alpha_{TT4}$, $o_{TT4}$, $d_{C2}$, $o_{C2}$ |

**Table 5**
End effector length calibration (corrections to the manually acquired values). Comparing calibration using plane constraints (h. plane – 1 horizontal plane, 2 h. planes – 2 horizontal planes, all planes), self-contact calibration, combination of planar constraints and self-contact (all) with self-observation only (on different datasets – e.g., S-O ($D^{hp}$) – self-observation calibration on a dataset from touching a horizontal plane). No cameras / Cam. calib / Fixed cameras — see text. Mean and standard deviation over 20 repetitions is displayed.

| Calibrating end effector length – corrections | | | |
|---|---|---|---|
| [mm] | No cameras | Cam. calib. | Fixed cameras |
| 1 h. plane | 10 ± 4 | 12 ± 2 | 3.32 ± 0.26 |
| 2 h. planes | 14 ± 2 | 15 ± 2 | 4.84 ± 0.11 |
| all planes | 13 ± 3 | 10.2 ± 0.2 | 4.34 ± 0.07 |
| self-contact | 4.31 ± 0.05 | 4.21 ± 0.04 | 4.12 ± 0.04 |
| all | 4.31 ± 0.04 | 4.25 ± 0.04 | 3.89 ± 0.04 |
| S-O ($D^{hp}$) | – | 8 ± 18 | 3.11 ± 0.24 |
| S-O ($D^{planes}$) | – | 10.17 ± 0.25 | 4.38 ± 0.11 |
| S-O ($D^{whole}$) | – | 6.25 ± 0.07 | 5.95 ± 0.10 |
| Calibrating end effector orientation – corrections | | | |
| [rad] $10^{-2}$ | No cameras | Cam.calib. | Fixed cameras |
| 1 h. plane | – | 2.96 ± 0.11 | 4.25 ± 0.23 |
| 2 h. planes | – | 3.29 ± 0.09 | 5.76 ± 0.21 |
| all planes | – | 3.28 ± 0.09 | 3.69 ± 0.17 |
| self-contact | – | 7.04 ± 0.09 | 1.06 ± 0.13 |
| all | – | 6.34 ± 0.08 | 2.20 ± 0.08 |
| S-O ($D^{hp}$) | – | 3.06 ± 0.15 | 4.17 ± 0.38 |
| S-O ($D^{planes}$) | – | 3.23 ± 0.13 | 3.60 ± 0.12 |
| S-O ($D^{whole}$) | – | 6.40 ± 0.09 | 2.39 ± 0.09 |

In the self-contact scenario, we compared the case where we assume that the length and offset for the left arm end effector are known and the case where we calibrate both EE1 and EE2 with the assumption that both end effectors have the same length—this is necessary to avoid compensation of the length of one end effector by the other end effector. In Table 5, we show results for the case when both end effectors lengths are calibrated simultaneously.

Table 5, the top part, summarizes the corrections to the nominal values of the length of the end effector achieved by individual calibration approaches. The most consistent estimates (mean corrections over 20 repetitions) across individual approaches are achieved with fixed precalibrated cameras, where the range of estimates is from 3.89 mm (all) to 4.84 mm (2 horizontal planes) and from 3.11 mm to 5.95 mm for the self-observation approach trained on three different datasets.

The quality of the individual estimates can also be evaluated with respect to the standard deviation (s.d.) across 20 repetitions. For planar constraints without cameras, we get a very high standard deviation (around 4 mm) indicating poor estimates. On the contrary, using the self-contact approach even without cameras results in standard deviation around 0.05 mm, which is comparable to the standard deviation achieved when camera information is included (either precalibrated cameras or cameras calibrated together with end effector length). The addition of cameras also improves consistency of end effector length

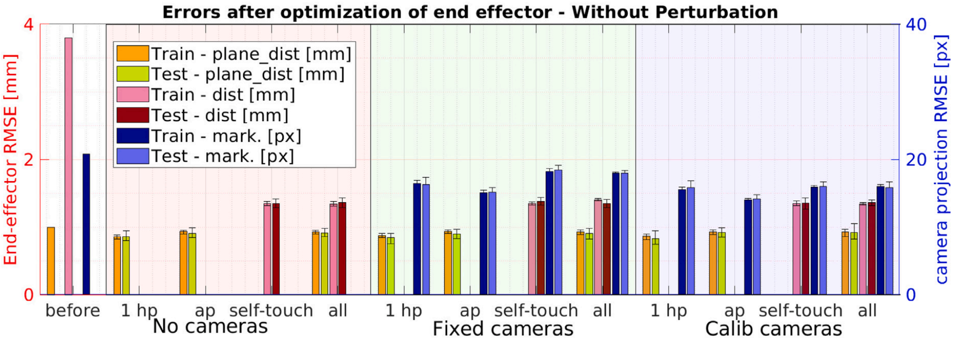


**Fig. 17.** RMS errors – end effector calibration. Distances in 3D (left $y$-axis, $mm$); camera reprojection (right $y$-axis, $px$). *plane_dist* – Euclidean distance from a plane, *dist* – Eucl. distance between end effectors in self-contact, *mark.* – error of individual markers reprojected to cameras. *No cameras* – only planar constraints or self-touch; *Fixed cameras* – precalibrated but fixed cameras; *Calib cameras* – precalibrated cameras with additional optimization. Error on both training (*Train*) and testing (*Test*) dataset shown. *before* – nominal parameters, *hp* – 1 horizontal plane, *ap* – all planes (2 horizontal, 1 vertical), *all* – self-contact + planar constraints.

estimates by planar constraints (standard deviation drops from 4 to 0.2 mm for calibrated cameras and to 0.07 mm for fixed cameras). We also evaluated the calibration using only self-observation information. When we used precalibrated cameras on the whole dataset, we achieve comparable results to other estimates. When we calibrate the cameras together with end effector length, the length of the end effector is overestimated and the standard deviation (based on the used training data) is also significantly higher.

We compare orientation corrections using only self-observation calibration (with cameras precalibrated on different subsets of datasets) or self-observation combined with other constraints (self-contact, planar constraints) – see Table 5, bottom part. The corrections to manually measured parameters of the orientation vary between 0–0.06 rad (0°–3.4°).

The resulting RMS errors (see Section 4.9) for end effector calibration in mm and px are shown in Fig. 17. For all variants of the optimization problem, both mean *dist* RMSE and *markers* RMSE decreased compared to the nominal end effector value (Fig. 17). For the variant *all* and *Fixed cameras*, the distance RMSE decreased from 3.8 mm before calibration to 1.40 mm after calibration and markers RMSE decreased from 20.8 px before to 18.0 px. Without cameras, we achieved slightly better results for self-contact distance RMSE (1.34 mm), but in this case the end effector orientation is not calibrated—resulting in higher error in camera reprojection. For the case with calibrated cameras, the resulting RMSE are slightly smaller (mean self-contact distance 1.34 mm and camera reprojection RMSE 16.0 px) than for the case with fixed cameras. The calibration resulted in the estimation of end effector length and orientation—the corrections of these parameters are listed in Table 5.

For the end effector parameters, we do not have any reference value apart from manual measurements (laser tracker calibration cannot be applied due to the retroreflector placement — see Fig. 3). However, the corrections found using the whole dataset and the combination of all chains (self-touch + planar constraints + cameras) were reliable enough. Consistent adaptation was observed for the length. Hence, for subsequent calibration of the robot arm kinematics, we used the end effector length estimate of 35.4 cm. For the orientation, no significant adaptation was arrived at and the nominal value was kept for the remaining experiments (see Table 1).

To validate the selection of the end effector length—which will be used in what follows as an initial value of this parameter—we performed two additional experiments. First, we systematically varied the initial end effector length parameter before calibration in the range from 10 to 70 mm and evaluated the RMS error after calibration on the self-contact dataset — see Fig. 18. Two solutions were found with minimal RMS error (35.38 cm and 61.4 cm), but one of them is not to be considered, as we do not expect errors bigger than 1 cm from manual measurement (35 cm). These two solutions can be explained by the nature of the self-contact as there are multiple possibilities arising from the geometrical consideration of self-contact (see Fig. 10, right).

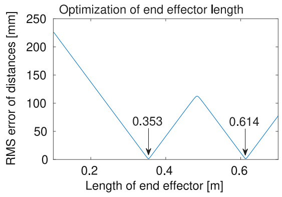


**Fig. 18.** RMS error on self-contact dataset for different values of end effector length.

Second, the sensitivity to perturbation of the initial end effector length for individual approaches was evaluated in the case with/without cameras and compared to the case without perturbation. The resulting corrections can be seen in Fig. 19. For the case with perturbation and without cameras (center), two solutions were found by *self-touch* calibration, corresponding to Fig. 17. When fixed precalibrated cameras are added, we achieve results with a low standard deviation for all calibration setups; for *self-touch* calibration, only one solution is selected. For *self-touch* and *all* conditions with fixed cameras, we achieve the same results both for perturbed initial value and non-perturbed.

### *5.2. Calibration of robot joint offsets by an external measurement device (reference)*

The whole dataset $D^{tracker}$ collected by the laser tracker (Leica) was used for right arm joint angle offsets calibration to create a reference value to our other calibration approaches (see Section 4.5). Parameters subject to calibration are listed in Table 4. First, we estimated the tracker position ($R$, $T$) and retroreflector DH parameters ($a$, $d$, and joint angle offset $o$) (link 7b in Table 1). Afterwards, we calibrated all offsets of the robot arm including the retroreflector DH parameters.

As can be seen in Fig. 20, left, RMSE improved with further calibration: from 6.08 mm before calibration to 3.03 mm after retroreflector calibration and to 2.64 mm after all offsets calibration. To achieve the best possible solution quality, we have not split the original dataset into training and testing part in this case. Instead, we evaluated the RMSE

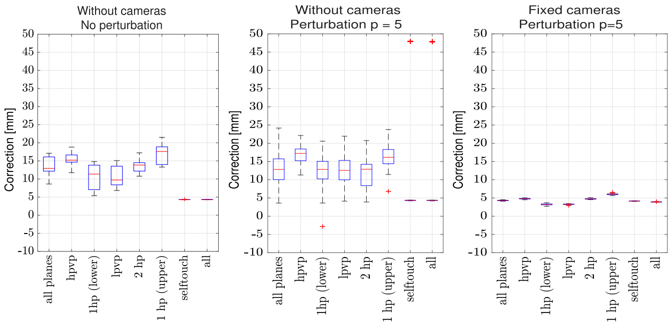


**Fig. 19.** Corrections of end effector length compared to nominal values: (left) no perturbation + no cameras, (center) perturbation + no cameras, (right) perturbation + fixed cameras. *all planes* – 2 horizontal, 1 vertical; *hpvp* – upper horizontal + vertical plane; *1hp (lower)* – lower horizontal plane; *lpvp* – lower horizontal + vertical plane; etc.; *all* – self-contact + all planes.

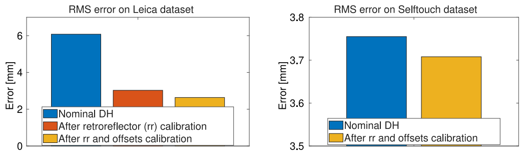


**Fig. 20.** (left) RMS errors comparison on the laser tracker dataset (*Nominal DH* – nominal parameters are used), after only retroreflector calibration, and after all right arm offsets including all DH parameters of the tracker retroreflector. (right) RMS errors comparison on self-contact dataset; parameters obtained by laser tracker calibration compared to RMS error achieved by Nominal DH parameters.

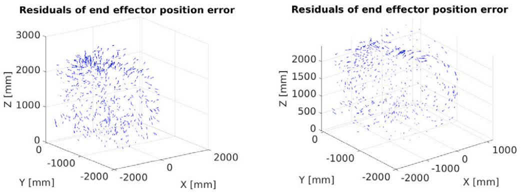


**Fig. 21.** Residual errors for individual datapoints before calibration using nominal parameters (left) and after calibration of all offsets of robotic arm including the tracker retroreflector full DH parameters (right).

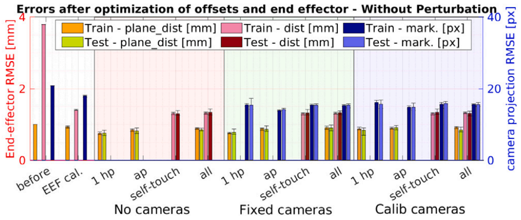


**Fig. 22.** RMS errors – right arm offsets and end effector calibration. Distances in 3D (left $y$-axis, $mm$); camera reprojection (right $y$-axis, $px$). *before* – nominal parameters, *EEF cal.* – after end effector calibration. For detailed legend see Fig. 17.

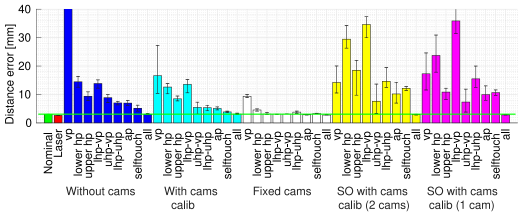


**Fig. 23.** Comparison of error on the testing dataset ($D^{tracker}$)– dataset used for laser tracker reference calibration. Error computed towards laser tracker retroreflector data. $x$ axis – individual datasets/methods used for finding robot kinematics parameters. We compare self-observation (*Self-obs.*) with one/both precalibrated cameras (on $D^{whole}$) calibrating robot offsets based on data from individual datasets, calibration using plane constraints (horizontal plane (*lower/upper*), 2 horizontal planes (*tables*), vertical plane (*vp*) and all planes (*ap*)), self-touch calibration (*self-touch*), and self-touch + planar constraints (*all*). Comparison when no cameras are used (*Without cams*), precalibrated (using self-observation on $D^{whole}$) but fixed cameras (*Fixed cams.*), and precalibrated cameras (on $D^{whole}$) being part of the calibration (*With cams calib*).

on a separate dataset (self-contact dataset $D^{st}$) using the parameters estimated by the laser tracker calibration and compared to RMSE for nominal parameters (Fig. 20, right). In this case, we used same parameters of the end effector for both methods and compared the error at the end of the 5th link (see Fig. 20). A slight improvement of RMSE can be observed also on this dataset: from 3.755 mm to 3.708 mm with the same uncalibrated end effector. The distribution of errors on the laser tracker dataset before and after calibration can be seen in Fig. 21.

The corrections for retroreflector parameters (compared to nominal parameters) $\{d_{EEL1}, a_{EEL1}\}$ are 0.72 mm, and 0.92 mm, respectively, and for offset parameters $\{o_{L1}, o_{U1}, o_{R1}, o_{B1}$, and $o_{ELL1}\}$ are $\{-1.43$, $-0.18$, $-0.59$, $-1.03$, and $4.123\}$ mrad, respectively.

### 5.3. Calibration of robot joint offsets – self-contained approaches

We compared the quality of the right arm joint angle offsets calibration, including the end effector length, by individual approaches. Self-contact approach (Section 4.1), planar constraints (using 1–3 planes; Section 4.2) and pure self-observation calibration (Section 4.3) were compared to the results for nominal parameters and the reference parameters acquired by the laser tracker measurement device (Section 5.2). For the contact-based methods (self-contact, planar constraints), we compared the results with no camera information, fixed precalibrated cameras, and cameras being part of the calibration process. Parameters subject to calibration are listed in Table 4. Depending on the particular method, additional parameters may be subject to calibration (e.g., parameters of the plane) (see corresponding subsections in Section 4).

RMSE after calibration of offsets and end effector length without perturbation can be seen in Fig. 22. The RMS error of distance in self-contact (in mm) drops from 3.80 mm before calibration to 1.40 mm after end effector calibration, and to 1.31 mm after all offsets calibration.

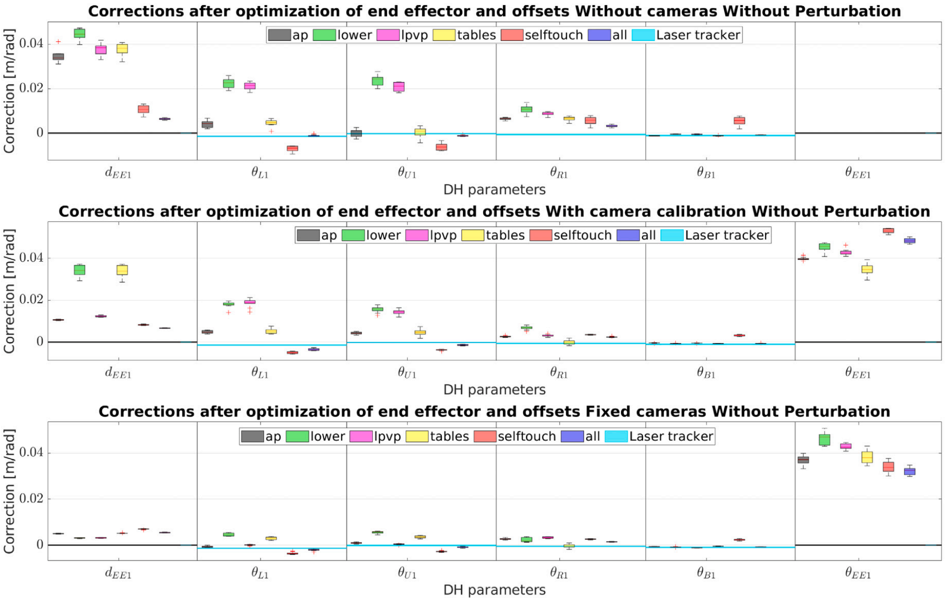


**Fig. 24.** Corrections of offsets and end effector parameters ($d_{EE1}$, $o_{EE1}$) after calibrating end effector and offsets of right arm without cameras (using only planar constraints or self-touch), with precalibrated but fixed cameras and with precalibrated cameras which are calibrated during the experiment. Offset of the end effector cannot be calibrated without cameras and nominal values are used. Laser tracker calibration (ground truth) values in turquoise.

Distance to plane (in mm) drops from 1.00 mm before calibration to 0.93 mmafter end effector calibration, and to 0.89 mm after all offsets calibration without cameras (0.91 mm with camera calibration and 0.92 mm with fixed cameras). Distance of reprojected markers in camera (in pixels) drops from 20.8 px before calibration to 18.0 px after end effector calibration to 15.3 px after all offsets calibration including camera calibration and to 15.6 px for fixed cameras calibration.

Corrections of the parameters to the nominal parameters are shown in Fig. 24. Corrections of the parameters are the smallest (also having the smallest standard deviation) compared to the values achieved by laser tracker calibration for the variant *all* with fixed precalibrated cameras, which means that we combined all of the chains (planar constraints, self-touch, self-observation) to optimize the offset parameters. The resulting correction for parameter $d_{EE1}$ is 5.46 ± 0.14 mm. The corrections for parameters {$o_{L1}$, $o_{U1}$, $o_{R1}$, $o_{B1}$, and $o_{EE1}$} are: {−2.10±0.38, −0.89±0.24, 1.48±0.13, −0.76±0.06, and 32.2±1.6} mrad, respectively. The corresponding corrections from laser tracker calibration for parameters {$o_{L1}$, $o_{U1}$, $o_{R1}$, and $o_{B1}$} are {−1.43, −0.18, −0.59, and −1.03} mrad, respectively.

### *5.4. Comparison of nominal DH parameters and former calibration/ calibration performed by laser tracker*

To evaluate the resulting offset parameters for the robot acquired through calibration using individual approaches, we conducted a comparison study on the laser tracker testing dataset $D^{tracker}$ (which covers the whole robot right arm workspace and not only the area where calibration using self-contained approaches was performed). To be able to perform such an evaluation, the transform to laser retroreflector, acquired from laser tracker calibration, was used for all compared calibration results. These results are shown in Fig. 23. Using the calibrated parameters by variant *all*, we can achieve better RMSE than nominal parameters—both in the case without cameras or with fixed precalibrated cameras. The resulting RMSE is 3.06 mm, 2.63 mm, 3.04 mm, 2.73 mm for nominal parameters, laser tracker, variant *all* without cameras, and variant *all* with precalibrated fixed cameras, respectively. Adding the option of camera calibration increases the RMSE for the variant *all* to 3.18 mm on this laser tracker testing dataset $D^{tracker}$ compared to the nominal parameters (3.06 mm).

The lowest RMS errors (for some setups lower than RMSE with nominal parameters, which is 3.06 mm) are achieved for different calibration approaches with fixed precalibrated cameras—the resulting RMSE are 9.3 mm, 4.5 mm, 3.8 mm, 3.31 mm, 2.84 mm, and 2.73 mm for vertical plane (*vp*), lower horizontal plane (*hp*), both horizontal planes (*tables*), self-contact (*selftouch*), all planes (*ap*), and variant *all*, respectively.

Pure self-observation (SO) highly depends on the quality of the training dataset. For the datasets from planar contact with one plane, we have only 250 datapoints and the resulting error is 29.4 mm for 2 cameras and 23.7 mm for 1 camera. When more data points are added—all planes (750 datapoints) or self-contact (566 datapoints)—the RMSE drops to 10.2 mm and 12.15 mm for all planes and self-contact, respectively. When $D^{whole}$ is used (1316 datapoints), the RMSE drops below RMSE achieved for nominal parameters (3.06 mm): to 2.90 mm for 2 cameras setup and 2.82 mm for 1 camera setup (still higher than variant *all* with fixed cameras – 2.73 mm).

### *5.5. Calibration of robot joint offsets — sensitivity to perturbation*

To evaluate the sensitivity of calibration approaches to perturbation, we run the experiment with initial perturbed parameters ($p$ = 3) (see Section 4.8 for details). The results can be seen in Fig. 25. For fixed precalibrated cameras we achieve very similar results as without perturbation with comparable low standard deviation. For example, for the variant *all* with fixed precalibrated cameras, the resulting correction for parameter $d_{EE1}$ is 5.46 ± 0.14 mm without perturbation and 5.52 ± 0.14 mmwith perturbation. The corrections for offset parameters $\{o_{L1}, o_{U1}, o_{R1} o_{B1}, o_{EE1}\}$, and are: {−2.10±0.38, −0.89±

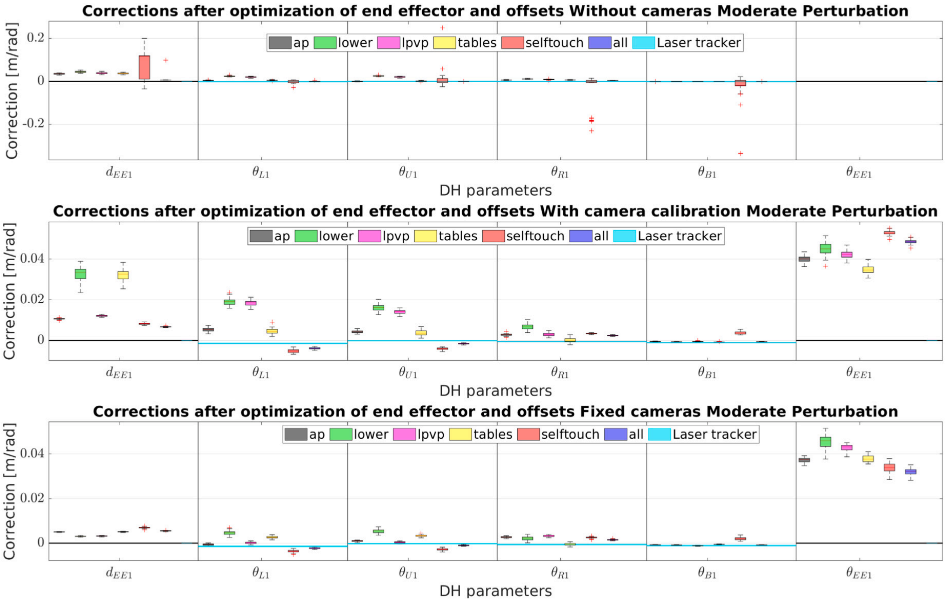


**Fig. 25.** Corrections of offsets and end effector parameters ($d_{EE1}$, $o_{EE1}$) after calibrating end effector and offsets of right arm without cameras (using only planar constraints or self-touch), with precalibrated but fixed cameras and with precalibrated cameras which are calibrated during the experiment. Offset of the end effector cannot be calibrated without cameras and nominal values are used. Laser tracker (Leica) calibration (ground truth) values in turquoise. Initial perturbation p = 3.

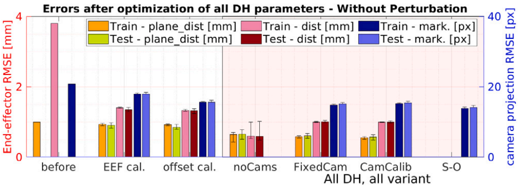


**Fig. 26.** RMS errors – right arm all DH parameters calibration. Distances in 3D (left $y$-axis, $mm$); camera reprojection (right $y$-axis, $px$). Results only for the variant *all* using dataset $D^{whole}$. *noCams* – only planar constraints + self-touch; *FixedCam* – precalibrated but fixed cameras; *CamCalib* – precalibrated cameras with additional optimization, *S-O* – self-observation only. *before* – nominal parameters, *EEF cal.* – after end effector calibration, *offset cal.* – after calibration of robot offsets + end effector. For detailed legend see Fig. 17.

$0.24, 1.48 \pm 0.13, -0.76 \pm 0.06, 32.2 \pm 1.6\}10^{-3}$ rad without perturbation, and $\{-2.24\pm0.28, -0.96\pm0.23, 1.55\pm0.20, -0.75\pm0.07, 32.1\pm1.3\}10^{-3}$ rad with perturbation, respectively. Without cameras, the corrections have higher standard deviation, especially for the 'self-touch' calibration. For all perturbation levels, the performance is stable (low error s.d.) in the case of fixed cameras.

### 5.6. Calibration of all DH parameters

In Fig. 26, we show a comparison of RMSE after calibration of all DH parameters (see Table 4) by the method *all* for different setups (fixed cameras, calibrated cameras and no cameras). Results on both training and testing datasets are shown and compared to the case where nominal parameters are used (*before*), EEF length and orientation calibration by the variant *all* (EEF cal.) and right arm offset calibration by the variant *all* (offset cal.). The self-contact distance, plane distance, and camera reprojection RMSE drops from 1.33 mm, 0.92 mm, and 15.6 px, respectively, after offset calibration, to 1.00 mm, 0.58 mm, and 14.8 px after calibration of all DH parameters with calibrated cameras, and further to 1.00 mm, 0.55 mm, and 15.2 px after calibration with precalibrated fixed cameras. When cameras are not used (*noCams*), self-contact distance further drops to 0.59 mm and planar distance to 0.64 mm. In this case, the end effector offset is not calibrated and the camera reprojection error would be higher.

In Fig. 27, we show the resulting corrections of DH parameters (corrections are calculated as the difference from the nominal parameters) for the variant *all* when perturbed DH parameters were used as initial value (perturbation factor $p = 3$). We show initial values of perturbed DH parameters, the results of calibration by method *all* in the case without cameras (using only contact information from planar constraints and self-touch), with precalibrated cameras which are calibrated during the experiment, and with precalibrated but fixed cameras. In all compared cases, we reach a lower dispersion of the final values and these values are very close to the industrial nominal DH parameters. The best results with the lowest standard deviation and closest to the industrial nominal values are consistently reached by the method when precalibrated fixed cameras are used. When no cameras are used, we can see that the corrections of DH parameters for variant *all* are unrealistically far from the nominal parameters (e.g., for parameter $a_{B1}$ we get corrections around 0.2 m). When also camera calibration is employed, the corresponding DH parameters corrections are reasonably close to nominal parameters and the calibration method is able to even for initially perturbed parameters (up to 10 cm difference from the nominal parameters) reach reasonable DH parameters values (all corrections are under 5 mm). We also show the results of the parameters from the laser tracker for comparison. End effector parameters cannot be calibrated by laser tracker thanks to its placement (see Fig. 3). Therefore, the value is not changing from the initial parameter. For the end effector length, we visualize as the nominal parameter the correction to the initial end effector length value achieved by the former end effector calibration (Section 5.1).

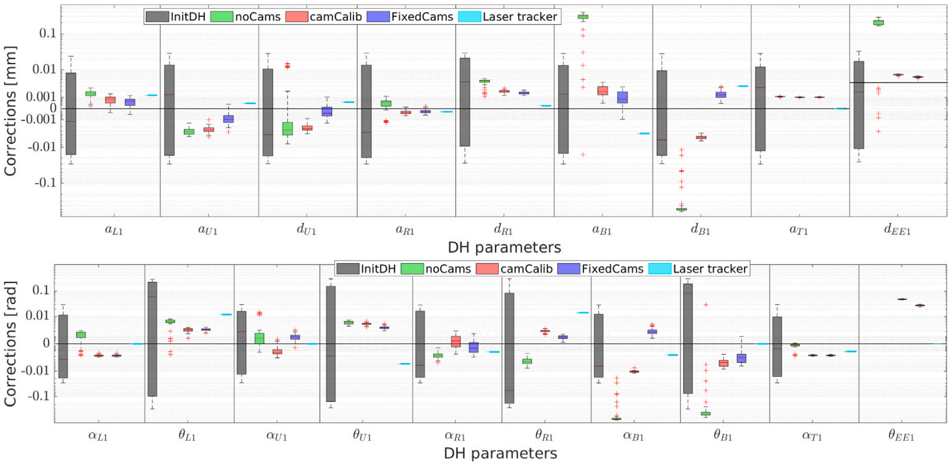


**Fig. 27.** Corrections of all DH parameters for the right arm by the variant *all* with initial perturbation ($p = 3$). Results without cameras (planar constraints + self-touch, *noCams*), with precalibrated cameras which are calibrated during the experiment (*camCalib*), and with precalibrated but fixed cameras (*FixedCams*). Offset of the end effector cannot be calibrated without cameras — nominal values are used. *Laser tracker* calibration (reference) values in tyrquis. Results shown over 50 repetitions.

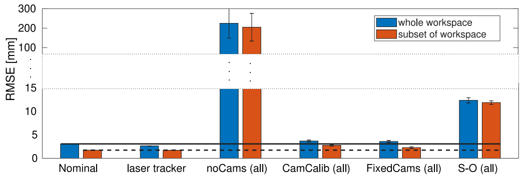


**Fig. 28.** Comparison of RMSE for all DH calibration on the whole laser tracker dataset ($D^{tracker}$). Results are shown for the whole laser tracker dataset (red) and for the subset of the robot workspace which corresponds to the area where other self-contact methods were trained (blue). Results for the calibration by the variant *all* without cameras (*noCams*), with fixed precalibrated cameras (*FixedCams*) and cameras being calibrated (*CamCalib*) is compared to the self-observation approach (*S-O*) trained on the dataset $D^{whole}$, *Nominal* parameters and the calibration by *laser tracker*. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

In Fig. 28, we show the resulting RMSE of our calibration contact-based approach on the testing laser tracker dataset $D^{tracker}$. The results for self-contact approach with fixed precalibrated cameras and with cameras calibrated as a part of the calibration approach are compared to self-observation approach where only camera information is used. All approaches are trained on the same $D^{whole}$ dataset. Comparison to the RMSE when nominal parameters and when DH parameters from laser tracker calibration are used, is provided. We show results for the whole laser tracker dataset and for the subset of the dataset which corresponds to the area where self-contact and self-observation approaches were calibrated. On the whole dataset, RMSE is $\{3.06 \pm 0, 2.63 \pm 0, 224 \pm 77, 3.68 \pm 0.20, 3.58 \pm 0.21,$ and $12.36 \pm 0.55\}$ mm for nominal parameters calibration (*Nominal*), *laser tracker*, self-contact calibration without cameras (*noCams (all)*), self-contact with cameras being part of calibration (*CamCalib (all)*), self-contact calibration with fixed precalibrated cameras (*FixedCams (all)*), and self-observation approach (*S-O (all)*), respectively. On the subset of the dataset, RMSE is $\{1.73 \pm 0, 1.71 \pm 0, 205 \pm 71, 2.75 \pm 0.17, 2.28 \pm 0.17,$ and $11.88 \pm 0.38\}$ mm for nominal parameters calibration (*Nominal*), *laser tracker*, self-contact calibration without cameras (*noCams (all)*), self-contact with cameras being part of calibration (*CamCalib (all)*), self-contact calibration with fixed precalibrated cameras (*FixedCams (all)*), and self-observation approach (*S-O (all)*), respectively. We can see that including self-contact information as a part of calibration provides significantly better results than using pure self-observation approach.

### 5.7. Observability analysis of individual approaches

We evaluated the observability indices $O_1$ and $O_4$ for the different combinations of calibration approaches and parameters subject to calibration. An overview is shown in Fig. 29 with $O_1$ on top and $O_4$ at the bottom. Please refer to Section 4.7 for details about how the indices are calculated. The bar groups on the $x$-axis correspond to the parameters subject to calibration and the number of parameters (columns of the identification Jacobian matrix **J**). Calibration using the laser tracker adds two additional parameters pertaining to the retroreflector placement (see Table 4). Color coding of individual bars marks the calibration approach with the no. data points in a given dataset without cameras/with cameras (rows of **J**). Due to the variable size of **J**, comparisons are to be made with caution. As expected, for the cases without cameras (*noCams*), there is a trend that with the number of calibrated parameters increasing (from *ee-noCams*, over *offsets-noCams*, to *allDH-noCams*), the observability indices are lower. When self-observation as a calibration method and corresponding camera extrinsic parameters are added (*calibCams*), observability indices $O_1$ in general significantly increase. Observability index $O_4$ increases after adding camera chains to calibration for the *allDH* calibration case, but drops for end effector length calibration—adding new parameters to be calibrated (columns of **J**) outweighs the effect of additional data from self-observation (rows of **J**). Comparing situations where the dataset and hence **J** has a similar size, we see that *horizontal plane* was more effective than *vertical plane* in our setup and that *selftouch* slightly outperformed combinations of all planar constraints (*all planes*). This is largely consistent with the RMSE errors after calibration (Fig. 23). Finally, for calibrating all DH parameters, the inclusion of cameras seems necessary.

Note that the observability analysis is also affected by measurement noise that may increase the effective rank of the Jacobian matrix. We assume that the measurement noise in this case is sufficiently small (experimental verification is presented at [3]).

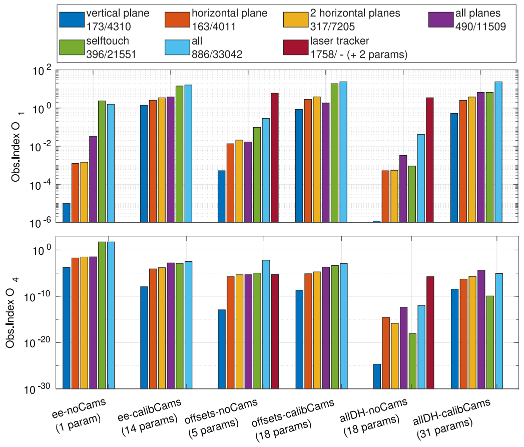


**Fig. 29.** Observability indices $O_1$ (top) and $O_4$ (bottom). Groups on the $x$-axis correspond to parameters subject to calibration (*ee* – end effector length; *offsets* – $o_i$ of individual links; *allDH* – all DH parameters; *noCams / calibCams* – without/with the "self-observation chains"; Table 4 for details). Color coding of individual bars marks the calibration approach and dataset (no. data points without cameras/with cameras).

## 6. Conclusion

Using a dual-arm industrial robot with force/torque sensing and cameras, we presented a thorough experimental comparison of geometric kinematic calibration using "self-contained approaches" using sensors on the robot—self-contact, planar constraints, and self-observation—with calibration using an external laser tracker. The main findings are summarized below.

First, we studied estimation of the kinematic parameters of a new tool—a custom end effector (Section 5.1). To calibrate the tool length, self-contact alone proved effective; planar constraints or self-observation alone did not perform as well in isolation and improved only in synergy with one of the other approaches/datasets. Testing RMS errors of approximately 1 to 1.5 mm on the part of the workspace where calibration was performed were achieved. For orientation of the tool, the addition of cameras (self-observation) was needed. Combining different methods/kinematic chains proved effective, supported also by the observability analysis (Section 5.7 and Fig. 29).

Second, we analyzed the performance of "daily calibration": the joint angle offsets of a complete robot arm were added to the tool calibration (Section 5.3). While the end effector calibration is responsible for most of the error in the workspace, further improvement from the offset calibration is achieved with all methods. To assess whether the calibration is effective only locally—on the particular dataset—or whether better kinematic parameters of the robot were learned, a comparison on an independent dataset covering the whole workspace (laser tracker calibration) was performed (Section 5.4 and Fig. 23). Corrections to robot parameter values were also analyzed. Importantly, all self-contained approaches except for the planar constraints in isolation were able to achieve good results, sometimes outperforming the robot nominal parameters. Self-contact alone proved the most effective; the best results were achieved using a combination of datasets and methods. Self-contact was further improved by the addition of planar constraints. Self-observation alone was sensitive to the size of the dataset but was effective in combination with contact-based methods; the best results were achieved using sequential calibration: camera extrinsics were calibrated first, followed by robot arm kinematics and tool calibration. Additionally, similar results were achieved when perturbing the initial parameter estimates (Section 5.5).

Third, we performed a complete geometric kinematic calibration of one robot arm plus the tool (all DH parameters) using different methods (Section 5.6). Good results are achieved on the individual datasets (Fig. 26), but further analysis reveals that the calibration suffered from overfitting to the particular dataset/part of the robot workspace. On an independent dataset, the performance of nominal parameters was not matched (Fig. 28). We also found that when all DH parameters are subject to calibration, the need for the synergy of different approaches increases, as testified by the unrealistic parameter corrections with the contact-only (noCams) approaches in Fig. 27 and the observability analysis (Fig. 29, *allDHnoCams*) or the self-observation only approach in Fig. 28. Like previously, the sequential calibration—first cameras, then robot kinematics—gives the best results. Although the performance of nominal or laser tracker calibration parameters on the whole workspace could not be matched (Fig. 28), the performance of the combination of self-contained approaches—also in the case of initial parameter perturbation—is reasonable (less than 4 mm error) and for a less accurate platform like a service robot may suffice.

Fourth, the comprehensive dataset collected is made publicly available [3] and can be used for additional analyses. This constitutes an additional contribution of this work.

## 7. Discussion and future work

Geometric kinematic calibration of industrial robots is usually performed using external metrology—measurement arms (e.g., [4]) or contactless measurement systems like laser trackers [5–8]. Newman et al. [7] calibrated a Motoman P-8 robot using an SMX laser tracker, improving the RMS error from 3.595 mm to 2.524 mm. Specifically related to the setup used in this work, our platform was previously calibrated using two different methods: (1) Redundant parallel Calibration and measuring Machine (RedCaM) by Beneš et al. [11], Volech et al. [12], and (2) Leica laser tracker. Petrík and Smutný [13] reviewed the precision of these methods using a linear probe sensor. Based on a dataset of 43 different poses with touching end effectors, they calculated the mean error as 0.67 (range 2.92) mm on CAD model, 0.54 (range 2.55) mm on Leica based calibration and 2.45 (range 9.92) mm on RedCaM based calibration. Other approaches—see [8] and additional references cited therein—usually achieve sub-millimeter accuracy. It was not our goal to directly compete with these works; instead, our aim was to assess the potential of automatic self-contained kinematic calibration: using sensors on the robot and avoiding the need for external metrology. We could demonstrate that such self-contained approaches—even if the initial robot parameters are perturbed—can yield less than 4 mm position errors over the robot workspace. The accuracy increases when a combination or synergy of these approaches (e.g., self-contact and self-observation) is exploited. We chose our platform, an industrial robot, out of convenience and to have a stable enough plant that can provide stationary results. However, we see the main application area in collaborative and service robotics. These platforms are typically more lightweight, flexible, and less precise and they may be often redeployed, their kinematic configuration changed etc. At the same time, they often come with a rich set of inexpensive but increasingly powerful sensors. Cameras and means to detect physical contact are becoming common. All these factors pave the way for automatic multisensorial self-contained calibration as demonstrated here.

We provide not only a thorough experimental investigation, but also a conceptual contribution in that we develop methods how to combine several of the methods into a single cost function. Further, unlike Birbach et al. [2] who claim that simultaneous calibration using all the available sensors is advantageous, we consistently found sequential calibration to perform best: camera calibration followed by robot kinematics. Conversely, calibrating robot kinematics together with camera extrinsics simultaneously was not as successful. Furthermore, if camera extrinsics were not precalibrated and inaccurate nominal parameters were used, the optimization often converged to physically impossible local minima. One known limitation of closed-loop calibration approaches—those relying on physical constraints on the end effector position or orientation—is that the set of poses is limited, which may affect the identifiability of parameters [1]. This holds pretty much for all self-contained calibration—self-contact or self-observation—as these also naturally constrain the set of robot configurations available. This has also been the case in this study. Better coverage of individual joint ranges within the approaches used here—in particular for the planar constraints—would further improve the results and yield better generalization to other parts of the workspace. Combining different methods / kinematic chains significantly mitigates this problem.

It should also be noted that although we used specially designed end effectors that were developed to combine self-contact (acting as a sphere) and self-observation (with flat tiles to host fiducial markers), the methods developed here have wider applicability. Contact can occur at any part of the robot provided that the link can be represented in the kinematic model; fiducial markers can be placed on any robot part or avoided altogether by tracking parts of the robot directly (e.g., [2,17,18]).

There are several directions for future work. First, when combining several calibration approaches into a single cost function (Section 4.4), the errors obtained from the different components could be scaled by coefficients that are inversely proportional to their uncertainty. We attempted to acquire such coefficients using repeatability measurements, but failed to obtain estimates that would reflect the true uncertainty associated with different approaches and possibly individual data points. Additional measurements of the uncertainty of individual components (beyond Section 4.10) and their propagation would be required. Thus, all components were weighted equally in this work, but this can be changed in the future.

Second, the methods presented here can be extended to exploit the existing sensors differently or to incorporate additional sensory modalities. For example, the two cameras in our setup were not used explicitly as a stereo head. Instead of reprojecting the end effector into the 2 image frames, one could also project the observed position of the end effector in image coordinates of both eyes (pixel $(u, v)$) to 3D space ($X^{eye}$) (similar to [17,56]). The self-contact approach would yield more than a 1-dimensional constraint in case the contact position (and hence 3 components of the pose) could be measured such as when using an artificial electronic skin [31,32,35]. Inertial sensors—in the robot head [2] or distributed on the robot body [57,58]—could be also added. Finally, one could also calibrate both manipulators simultaneously.

Third, for online recalibration to be performed repeatedly, the number of poses / data points needed can be reduced by employing intelligent pose selection (e.g., [59–62]).

Fourth, the standard calibration method using non-linear least squares optimization (Levenberg–Marquardt algorithm) can be compared with filtering approaches [18,26] or with methods that pose fewer assumptions on the initial model available (e.g., [63]).

## Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.rcim.2021.102250.

## Acknowledgments

We would like to thank Martin Hoffmann for designing and fabricating several versions of the custom end effectors, with assistance from Tomáš Báča for 3D printing. We are also indebted to CIIRC CTU for support in using the robot and the Leica absolute tracker (Libor Wágner, Vladimír Smutný, and Václav Hlaváč). Tomáš Svoboda provided general support, consultation, and pointer to the article [49].

## Funding

This work was supported by the Czech Science Foundation (GA ČR), project EXPRO (no. 20-24186X); T.P. was supported by the European Regional Development Fund under project Robotics for Industry 4.0 (no. CZ.02.1.01/0.0/0.0/15_003/0000470) and EU H2020 SPRING Project (no. 871245).

## References

[1] J. Hollerbach, W. Khalil, M. Gautier, Model identification, in: B. Siciliano, O. Khatib (Eds.), Springer Handbook of Robotics, second ed., Springer, 2016, pp. 113–138.

[2] O. Birbach, U. Frese, B. Bäuml, Rapid calibration of a multi-sensorial humanoid's upper body: An automatic and self-contained approach, Int. J. Robot. Res. 34 (4–5) (2015) 420–436.

[3] K. Stepanova, J. Rozlivek, F. Puciow, P. Krsek, T. Pajdla, M. Hoffmann, Self-calibration Dataset, 2020, http://karlastepanova.cz/publications/self-calibration-industry/.

[4] L.S. Ginani, J.M.S. Motta, Theoretical and practical aspects of robot calibration with experimental verification, J. Braz. Soc. Mech. Sci. Eng. 33 (1) (2011) 15–21.

[5] I.-C. Ha, Kinematic parameter calibration method for industrial robot manipulator using the relative position, J. Mech. Sci. Technol. 22 (6) (2008) 1084.

[6] H.-N. Nguyen, J. Zhou, H.-J. Kang, A new full pose measurement method for robot calibration, Sensors 13 (7) (2013) 9132–9147.

[7] W.S. Newman, C.E. Birkhimer, R.J. Horning, A.T. Wilkey, Calibration of a Motoman P8 robot based on laser tracking, in: Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065), Vol. 4, IEEE, 2000, pp. 3597–3602.

[8] A. Nubiola, I.A. Bonev, Absolute calibration of an ABB IRB 1600 robot using a laser tracker, Robot. Comput.-Integr. Manuf. 29 (1) (2013) 236–245.

[9] M. Švaco, B. Šekoranja, F. Šuligoj, B. Jerbić, Calibration of an industrial robot using a stereo vision system, Procedia Eng. 69 (2014) 459–463.

[10] J.-S. Hu, J.-J. Wang, Y.-J. Chang, Kinematic calibration of manipulator using single laser pointer, in: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2012, pp. 426–430.

[11] P. Beneš, M. Valášek, Z. Šika, V. Bauma, V. Hamrle, Experiments with Redundant Parallel Calibration and Measuring Machine RedCaM, Tech. Rep., University of West Bohemia, 2007.

[12] J. Volech, L. Mráz, Z. Šika, M. Valášek, Concepts of robot accuracy enhancement by integrated redundant measurements., Bull. Appl. Mech. 9 (33) (2013).

[13] V. Petrík, V. Smutný, Comparison of Calibrations for the CloPeMa Robot, Tech. Rep., Czech Technical University in Prague, 2014.

[14] D.J. Bennett, J.M. Hollerbach, Autonomous calibration of single-loop closed kinematic chains formed by manipulators with passive endpoint constraints, IEEE Trans. Robot. Autom. 7 (5) (1991) 597–606.

[15] M. Hersch, E. Sauser, A. Billard, Online learning of the body schema, Int. J. Hum. Robot. 5 (2008) 161–181.

[16] R. Martinez-Cantin, M. Lopes, L. Montesano, Body schema acquisition through active learning, in: Proc. Int. Conf. on Robotics and Automation, ICRA, 2010.

[17] S.R. Fanello, U. Pattacini, I. Gori, V. Tikhanoff, M. Randazzo, A. Roncone, F. Odone, G. Metta, 3D stereo estimation and fully automated learning of eye-hand coordination in humanoid robots, in: 2014 IEEE-RAS Int. Conf. on Humanoid Robots, HUMANOIDS '14, 2014.

[18] P. Vicente, L. Jamone, A. Bernardino, Online body schema adaptation based on internal mental simulation and multisensory feedback, Front. Robot. AI 3 (2016) 7.

[19] D.J. Bennett, J.M. Hollerbach, Self-calibration of single-loop, closed kinematic chains formed by dual or redundant manipulators, in: Decision and Control, 1988., Proceedings of the 27th IEEE Conference on, IEEE, 1988, pp. 627–629.

[20] A. Joubair, I.A. Bonev, Kinematic calibration of a six-axis serial robot using distance and sphere constraints, Int. J. Adv. Manuf. Technol. 77 (1–4) (2015) 515–523.

[21] M.C. Koval, M. Klingensmith, S.S. Srinivasa, N.S. Pollard, M. Kaess, The manifold particle filter for state estimation on high-dimensional implicit manifolds, in: Robotics and Automation (ICRA), 2017 IEEE International Conference on, IEEE, 2017, pp. 4673–4680.

[22] X. Xu, D. Zhu, J. Wang, S. Yan, H. Ding, Calibration and accuracy analysis of robotic belt grinding system using the ruby probe and criteria sphere, Robot. Comput.-Integr. Manuf. 51 (2018) 189–201.

[23] M. Ikits, J.M. Hollerbach, Kinematic calibration using a plane constraint, in: Proceedings of International Conference on Robotics and Automation, Vol. 4, IEEE, 1997, pp. 3191–3196.

[24] H. Zhuang, S.H. Motaghedi, Z.S. Roth, Robot calibration with planar constraints, in: Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C), Vol. 1, IEEE, 1999, pp. 805–810.

[25] A. Joubair, I.A. Bonev, Non-kinematic calibration of a six-axis serial robot using planar constraints, Precis. Eng. 40 (2015) 325–333.

[26] R. Zenha, P. Vicente, L. Jamone, A. Bernardino, Incremental adaptation of a robot body schema based on touch events, in: Proc. Joint IEEE Int. Conf. Develop. Learn. Epigenetic Robot, 2018.

[27] R. Khusainov, A. Klimchik, E. Magid, Humanoid robot kinematic calibration using industrial manipulator, in: Mechanical, System and Control Engineering (ICMSC), 2017 International Conference on, IEEE, 2017, pp. 184–189.

[28] R. Wang, A. Wu, X. Chen, J. Wang, A point and distance constraint based 6R robot calibration method through machine vision, Robot. Comput.-Integr. Manuf. 65 (2020) 101959.

[29] C. Bartolozzi, L. Natale, F. Nori, G. Metta, Robots with a sense of touch, Nature Mater. 15 (9) (2016) 921–925.

[30] R. Dahiya, D. Akinwande, J.S. Chang, Flexible electronic skin: From humanoids to humans [scanning the issue], Proc. IEEE 107 (10) (2019) 2011–2015.

[31] A. Roncone, M. Hoffmann, U. Pattacini, G. Metta, Automatic kinematic chain calibration using artificial skin: self-touch in the iCub humanoid robot, in: Robotics and Automation (ICRA), 2014 IEEE International Conference on, 2014, pp. 2305–2312.

[32] Q. Li, R. Haschke, H. Ritter, Towards body schema learning using training data acquired by continuous self-touch, in: Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on, IEEE, 2015, pp. 1109–1114.

[33] P. Mittendorfer, G. Cheng, Humanoid multimodal tactile-sensing modules, Robotics, IEEE Trans. 27 (3) (2011) 401–410.

[34] O. Limoyo, T. Ablett, F. Marić, L. Volpatti, J. Kelly, Self-calibration of mobile manipulator kinematic and sensor extrinsic parameters through contact-based interaction, in: 2018 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2018, pp. 1–8.

[35] K. Stepanova, T. Pajdla, M. Hoffmann, Robot self-calibration using multiple kinematic chains – a simulation study on the iCub humanoid robot, IEEE Robot. Autom. Lett. 4 (2) (2019) 1900–1907.

[36] R.A. Boby, A. Klimchik, Combination of geometric and parametric approaches for kinematic identification of an industrial robot, Robot. Comput.-Integr. Manuf. 71 (2021) 102142.

[37] Z. Jiang, M. Huang, X. Tang, Y. Guo, A new calibration method for joint-dependent geometric errors of industrial robot based on multiple identification spaces, Robot. Comput.-Integr. Manuf. 71 (2021) 102175.

[38] Z. Fu, J.S. Dai, K. Yang, X. Chen, P. López-Custodio, Analysis of unified error model and simulated parameters calibration for robotic machining based on Lie theory, Robot. Comput.-Integr. Manuf. 61 (2020) 101855.

[39] G. Luo, L. Zou, Z. Wang, C. Lv, J. Ou, Y. Huang, A novel kinematic parameters calibration method for industrial robot based on Levenberg–Marquardt and Differential Evolution hybrid algorithm, Robot. Comput.-Integr. Manuf. 71 (2021) 102165.

[40] CloPeMa, CloPeMa: The robot, 2014, http://clopemaweb.felk.cvut.cz/the-robot/, (Accessed: 18 may 2018).

[41] M. Quigley, K. Conley, B.P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, A.Y. Ng, ROS: An open-source Robot Operating System, in: ICRA Workshop on Open Source Software, 2009.

[42] Leica geosystems, Leica absolute tracker AT402, 2020, https://www.hexagonmi.com/products/laser-tracker-systems/leica-absolute-tracker-at403, (Accessed: 20 June 2020).

[43] B. Böckem, D. Moser, S. Fuchs, Coordinate measuring device and method, (EP2980526B1) 2014.

[44] R.S. Hartenberg, J. Denavit, A kinematic notation for lower pair mechanisms based on matrices, J. Appl. Mech. 77 (2) (1955) 215–221.

[45] G. Bradski, The openCV library, Dr. Dobb's J. Softw. Tools (2000).

[46] F. Puciow, Automatic Self-Calibration from Self-Observation and Self-Touch on a Dual Arm Industrial Manipulator (Master's thesis), FEE CTU, 2018.

[47] K. Nickels, Hand-Eye Calibration for Robonaut, Tech. Rep., NASA Summer Faculty Fellowship Program Final Report, 2003.

[48] J. Rozlivek, L. Rustler, K. Stepanova, M. Hoffmann, Multisensorial Robot Calibration Framework and Toolbox, Github, 2021, https://github.com/ctu-vras/multirobot-calibration.

[49] K.S. Arun, T.S. Huang, S.D. Blostein, Least-squares fitting of two 3-D point sets, IEEE Trans. Pattern Anal. Mach. Intell. PAMI-9 (5) (1987) 698–700.

[50] J. Hollerbach, C. Wampler, The calibration index and taxonomy for robotic kinematic calibration methods, Int. J. Robot. Res. 15(6) (1996) 573–591.

[51] Y. Sun, J.M. Hollerbach, Observability index selection for robot calibration, in: 2008 IEEE International Conference on Robotics and Automation, IEEE, 2008, pp. 831–836.

[52] J.-H. Borm, C.-H. Menq, Experimental study of observability of parameter errors in robot calibration, in: Proceedings, 1989 International Conference on Robotics and Automation, IEEE, 1989, pp. 587–592.

[53] M.R. Driels, U.S. Pathre, Significance of observation strategy on the design of robot calibration experiments, J. Robot. Syst. 7 (2) (1990) 197–223.

[54] A. Nahvi, J.M. Hollerbach, V. Hayward, Calibration of a parallel robot using multiple kinematic closed loops, in: Proceedings of the 1994 IEEE International Conference on Robotics and Automation, IEEE, 1994, pp. 407–412.

[55] A. Nahvi, J.M. Hollerbach, The noise amplification index for optimal pose selection in robot calibration, in: Proceedings of IEEE International Conference on Robotics and Automation, Vol. 1, IEEE, 1996, pp. 647–654.

[56] H. Hirschmuller, Stereo processing by semiglobal matching and mutual information, IEEE Trans. Pattern Anal. Mach. Intell. 30 (2) (2008) 328–341.

[57] N. Guedelha, N. Kuppuswamy, S. Traversaro, F. Nori, Self-calibration of joint offsets for humanoid robots using accelerometer measurements, in: Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on, IEEE, 2016, pp. 1233–1238.

[58] P. Mittendorfer, G. Cheng, Open-loop self-calibration of articulated robots with artificial skins, in: Robotics and Automation (ICRA), 2012 IEEE International Conference on, IEEE, 2012, pp. 4539–4545.

[59] T. Li, K. Sun, Z.-w. Xie, H. Liu, Optimal measurement configurations for kinematic calibration of six-DOF serial robot, J. Cent. South Univ. Technol. 18 (3) (2011) 618–626.

[60] D. Daney, Y. Papegay, B. Madeline, Choosing measurement poses for robot calibration with the local convergence method and Tabu search, Int. J. Robot. Res. 24 (6) (2005) 501–518.

[61] J. Zhou, H.-N. Nguyen, H.-J. Kang, Selecting optimal measurement poses for kinematic calibration of industrial robots, Adv. Mech. Eng. 6 (2014) 291389.

[62] W. Wang, H. Song, Z. Yan, L. Sun, Z. Du, A universal index and an improved PSO algorithm for optimal pose selection in kinematic calibration of a novel surgical robot, Robot. Comput.-Integr. Manuf. 50 (2018) 90–101.

[63] P. Lanillos, G. Cheng, Adaptive robot body learning and estimation through predictive coding, in: Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on, 2018.

Appendix L

# Safe physical HRI: Toward a unified treatment of speed and separation monitoring together with power and force limiting

Authors' version of:

Svarny, P., Tesar, M., Behrens, J.K., and Hoffmann, M. (2019). Safe physical HRI: Toward a unified treatment of speed and separation monitoring together with power and force limiting. In *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, pages 7574-7581.

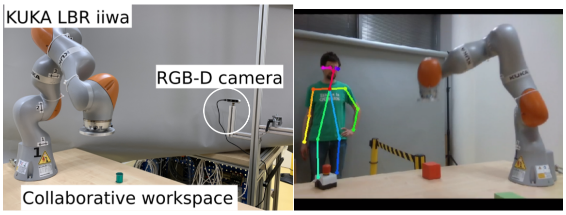
DOI: https://doi.org/10.1109/IROS40897.2019.8968463
Youtube video: https://youtu.be/-TC4WsdJJAM

Author contributions: The contribution of M. Hoffmann was 20%.

# Safe physical HRI: Toward a unified treatment of speed and separation monitoring together with power and force limiting

Petr Svarny, Michael Tesar, Jan Kristof Behrens, and Matej Hoffmann

***Abstract*— So-called collaborative robots are a current trend in industrial robotics. However, they still face many problems in practical application such as reduced speed to ascertain their collaborativeness. The standards prescribe two regimes: (i) speed and separation monitoring and (ii) power and force limiting, where the former requires reliable estimation of distances between the robot and human body parts and the latter imposes constraints on the energy absorbed during collisions prior to robot stopping. Following the standards, we deploy the two collaborative regimes in a single application and study the performance in a mock collaborative task under the individual regimes, including transitions between them. Additionally, we compare the performance under "safety zone monitoring" with keypoint pair-wise separation distance assessment relying on an RGB-D sensor and skeleton extraction algorithm to track human body parts in the workspace. Best performance has been achieved in the following setting: robot operates at full speed until a distance threshold between any robot and human body part is crossed; then, reduced robot speed per power and force limiting is triggered. Robot is halted only when the operator's head crosses a predefined distance from selected robot parts. We demonstrate our methodology on a setup combining a KUKA LBR iiwa robot, Intel RealSense RGB-D sensor and OpenPose for human pose estimation.**



(a) (b)

Fig. 1: Experimental setup – collaborative workspace. (a) External view. (b) Camera view with human keypoint extraction.

## I. Introduction

So-called "collaborative robots" (or "cobots"), i.e. robots that are safe when sharing the same (collaborative) workspace with human operators, represent a rising trend in robotics. However, their industrial application is limited by their performance—the reduced speed and limited payload in particular. Safe *physical Human-Robot Interaction* (pHRI) saw great development in the last decade, with the introduction of new safety standards [1], [2] and a rapidly growing market of cobots. However, it is a more recent attempt to enhance not only the safety of these robots but also their performance. This attempt to make collaborative robotics more attractive to the traditional industry is visible also in projects promoting the advancement in this field (see the COVR project[1] [3]).

Haddadin and Croft [4] provide a survey of pHRI. According to [2], there are two ways of satisfying the safety requirements when a human physically collaborates with a robot: (i) *Power and force limiting (PFL)* and (ii) *Speed and separation monitoring (SSM)*. For PFL, physical contacts with a moving robot are allowed but the forces / pressures / energy absorbed during a collision need to be within human body part specific limits. This translates onto lightweight structure, soft padding, no pinch points, and possibly introduction of elastic elements (see the series elastic actuators in Sawyer robot; [5] for a formal treatment of robots with flexible joints) on the robot side, in combination with collision detection and response relying on motor load measurements, force/torque or joint torque sensing. This is addressed by interaction control methods for this *post-impact* phase (see [5] for a recent survey). The performance of robots complying with this safety requirement in terms of payload, speed, and repeatability is limited.

Safe collaborative operation according to speed and separation monitoring prohibits contacts with a moving robot and thus focuses on the *pre-impact* phase: a *protective separation distance*, $S_p$, between the operator and robot needs to be maintained at all times. When the distance decreases below $S_p$, the robot stops [2].

In industry, $S_p$ is typically safeguarded using light curtains (essentially electronic versions of physical fences) or safety-rated scanners that monitor 2D or 3D zones (e.g., Pilz SafetyEYE). One can usually define a protection field (denoted "red" zone)—if an object is detected inside, the robot is brought to an immediate halt—and a warning field (called "yellow" zone) that may trigger a reduced maximum allowed robot speed. However, the flexibility of such setups is limited: the information is reduced to detecting whether an object of a certain minimum volume has entered one of the two predefined zones. Also, the higher the robot kinetic energy, the bigger is its footprint on the shop floor.

With increasing performance and falling prices of RGB-D sensors (RGB image + depth information), we can prototype collaborative scenarios using already available sensors (like Intel RealSense) and tools for human keypoint or skeleton

Petr Svarny and Matej Hoffmann are with the Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague. Michael Tesar and Jan K. Behrens with the Czech Institute of Informatics, Robotics, and Cybernetics of the Czech Technical University in Prague. (e-mail: petr.svarny@fel.cvut.cz; michael.tesar@cvut.cz; jan.kristof.behrens@cvut.cz, matej.hoffmann@fel.cvut.cz).

[1] http://safearoundrobots.com/

extraction from camera images [6], [7]. This combination permits real-time perception of the positions of individual body parts of any operators in the collaborative workspace. Deployment in real applications will depend on the development of safety-rated modules providing this functionality[2].

In this work, we take advantage of the keypoint information and follow [2] to deploy the two collaborative regimes (SSM and PFL) in a single application. The deployment of both regimes in a single scenario provides in our view the unique contribution of this work. The PFL regime prescribes different thresholds for the body parts of the operator and hence only with the keypoint information available can the body part specific limits be taken into consideration—demonstrated on the head keypoints here. We study the performance in a mock collaborative task under different settings like distances from robot base to individual keypoints, stopping or slowing down, and their transitions—the distances and speeds are based on [2] in our setup. We use a KUKA LBR iiwa collaborative robot, Intel RealSense RGB-D sensor and OpenPose for human pose estimation as shown in Fig. 1.

This article is structured into related work reviewed in the next section, followed by Materials and Methods, and Results. We close by Discussions and Conclusions.

## II. Related work

A functional solution for safe pHRI according to the speed and separation monitoring requirements will necessarily involve: (i) sensing of the human operators' as well as robot's positions (and speeds), (ii) a suitable representation of the corresponding separation distances, and (iii) appropriate responses of the machine (speed reduction / stop / avoidance maneuvers). On the perception side, tracking the robot parts in space tends to be relatively easy as accurate models of the machine as well as joint encoder readings are available and hence position (and possibly also orientation, speed, and acceleration) for the end-effector as well as other chosen keypoints can be readily obtained from forward kinematics. On the other hand, the perception of the human operators in the workspace is more challenging. Two key technologies have appeared that facilitate progress in this area: (i) compact and affordable RGB-D sensors and (ii) convolutional neural networks for human keypoint/skeleton extraction from camera images [6], [7], or full 3D human body reconstruction [8]. These technologies together—albeit currently not safety-rated—make it possible to perceive the positions of individual body parts of any operators in the collaborative workspace in real time. Alternative technologies include distributed wireless sensor networks that track operators who do not wear any devices [9] or proximity sensors distributed on the robot, usually part of electronic skins (e.g., Bosch APAS robot). The main benefit of all these solutions is their resolution—compared to mere zone monitoring—and hence reduction of the effective footprint of the robot.

Once the robot and human positions are obtained, their relative distances (and possibly speeds or time to collision) need to be evaluated. Euclidean distance is the most natural candidate and also one that appears in the safety norms. However, other representations have been proposed and may be better suited for the nature of the sensory data (like the depth space approach for RGB-D data [10], [11]) or for planning and control of the robot where the configuration space (joint space) of the robot can be used for representing both the robot body and the obstacles. Flacco et al. [11] provide an overview. Another key component is in what form are the robot and human body parts represented. Drawing on the results of the computer graphics community ([12] for a survey), this often takes the form of some collision primitives. These can be simple shapes like spheres [10] or more complex meshes [13] and can differ for the robot and the human: Zanchettin et al. [14] represent robot links as segments and humans as a set of capsules. Of course, for safety to be guaranteed, the whole body of both agents should be represented and considering only the robot end-effector does not suffice. Often, the "robot-centered" approach is taken—in the sense that the collision primitives are centered on the robot body and possibly dynamically shaped based on the current robot velocity [13], [14], [15], [16]. A biologically inspired approach relying on peripersonal space representation was presented in [17], [18].

Interaction control methods for the *post-impact* phase (see [5] for a survey) are not our focus here. We rely mainly on the information in [2] to calculate the speed our robot can run with while fulfilling the PFL regime criteria.

There is a large body of work dealing with motion planning and control in dynamic environments. In the face of dynamically appearing obstacles (the case in HRI scenarios), classical offline trajectory planning [19] has to be complemented by reactive strategies [20], [21]. This problem gives rise to new velocity-dependent formulations such as "velocity obstacles" [22] or "dynamic envelope" [23]. Recently, the approaches are somewhat closer to the "control" than to the "planning" community: the work of De Luca and Flacco ([10]; [24] deal with both pre-impact and post-impact control) or Zanchettin et al. [14] are good examples. In summary, researchers in robotics often find themselves developing compelling solutions for real-time obstacle avoidance, but these may require substantial tuning and the separation distance is often optimized rather than guaranteed (e.g., [15], [20]). There are notable exceptions like the work of Marvel [25] and Zanchettin et al. [14] that take the constraints imposed by the safety standards seriously. Regarding the PFL regime, Sloth and Petersen [26] recently presented a method to compute safe path velocities complying with [2]; Mansfeld et. al. [27] developed a "safety map" and use alternative, less conservative, collision limits derived from biomechanics impact data. Similarly, [28], [29] provide a treatment of robot control taking into account the energy dissipated in possible contacts with the operator.

The SSM part of our framework follows up on our previous work [18], [30], in which we take advantage of the

[2] For example , `https://www.veobot.com/`

keypoint extraction to monitor distances between individual parts of the human and robot body and exploit also the keypoint semantics to modulate the behavior. In this work, we make important steps in bringing these ideas to an industrial setting by moving to an industrial collaborative robot, adding the PFL regime, and illustrating how to determine all the relevant parameters in accordance with [2].

## III. Materials and Methods

### A. Robot platform

A 7 DoF industrial manipulator KUKA LBR iiwa 7 R800 was used. The robot operates either at full speed (up to 1 m/s for the end-effector) or reduced speed (0.42 m/s). As an additional low-level safety layer, the KUKA *Collision detection* based on external torque estimation was turned on.

### B. RGB-D camera

The camera was an Intel RealSense D435 RGB-D. We calibrate the robot and camera position through the ROS Hand-Eye calibration tool. The camera resolution is 848x480, and we use the RealSense short range presets[3].

### C. HRI setup

Our setup is illustrated in Fig. 1. A mock collaborative task has been staged: the robot performs a periodic operation. Operator periodically replaces one of the objects, entering the robot workspace, and is perceived by the camera. The robot responds appropriately (slow down or stop). The robot was placed on a fixed table while the RGB-D sensor was on a fixed position so that it can capture the whole robot workspace. The camera was fixed to a construction that was separate from the robot's platform to avoid tremors during the robot's movement. The setup was designed to minimize the chance of occlusions.[4]

### D. Software framework and robot control

A schematics of the overall framework is shown in Fig. 2. OpenPose (see Sec. III-E) finds human keypoints in pictures captured by the camera as orchestrated by a ROS node. The robot node consumes and produces information about the coordinate transformations. The relative distances are assessed in the peripersonal space module (*pps*) and fed into the robot controller to generate the appropriate response.

High-level control of the robot was done in the ROS node *move_robot*. We used the MoveIt! motion planning framework [32] to generate and execute the trajectories for our mock task. Our scenario additionally required speed modulation (stop, slow down, speed up) on the run which is not provided by Moveit! and we have implemented a custom solution for smoothly modulating the trajectories in joint space, compliant with the corresponding limits of the platform. In brief, we used cascaded robot control which masks system non-linearities and lets us see the robot as a system of seven double-integrators, which we control similarly to a saturation controller [33]. We distinguish:

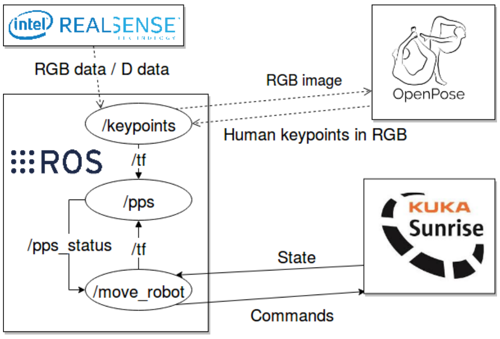


Fig. 2: Software architecture schematics.

(i) **Stopping motion.** The remaining trajectory of the robot is replaced by an alternative trajectory with a maximal deceleration for the fastest joint and relatively scaled deceleration for all other joints. The overall stopping time $t_e$ is dependent on the velocity of the joints $\dot{x}_j$ and the acceleration limits $a_{\text{j,min}} \leq \ddot{x}_j \leq a_{\text{j,max}}$, $t_{\text{stop},j}$ denotes the minimal stopping time for a joint $j$:

$$t_e = \max_{j \in Joints} t_{\text{stop},j} \tag{1}$$

$$t_{\text{stop},j} = \begin{cases} \dot{x}_j < 0 & \frac{0 - \dot{x}_{\text{j,ref}}}{a_{\text{j,max}}} \\ \dot{x}_j \geq 0 & \frac{0 - \dot{x}_{\text{j,ref}}}{a_{\text{j,min}}} \end{cases} \tag{2}$$

The worst-case run-time of the stopping trajectory calculation $t_{\text{calc}} \leq 0.02\ s$ was determined empirically. When the stop signal arrives, the earliest future state (with $t \geq t_{\text{now}} + t_{\text{calc}}$) along the current trajectory is selected and used as reference state $\mathbf{x}_{\text{ref}}$ for calculations.

$$\dot{x}_j(0) = b_{j,1} = \dot{x}_{\text{j,ref}} \tag{3}$$

$$\dot{x}_j(t_e) = b_{j,1} + b_{j,2} t_e = 0 \longrightarrow b_{j,2} = \frac{\dot{x}_{\text{j,ref}}}{2 t_e} \tag{4}$$

$$x_j(t) = \underbrace{x_{\text{j,ref}}}_{b_{j,0}} + \underbrace{\dot{x}_{\text{j,ref}}}_{b_{j,1}} t + \underbrace{\frac{\dot{x}_{\text{j,ref}}}{2 t_e}}_{b_{j,2}} t^2 \tag{5}$$

To facilitate the full breaking potential, we use polynomials (with parameters $b_{j,0}$, $b_{j,1}$ and $b_{j,2}$) of degree two to describe the joint positions. Hence, the velocities $\dot{x}_{\text{j}}$ are linear with the maximum deceleration for at least one joint. This breaking behavior yields the shortest stopping time possible, but will for general trajectories slightly deviate from the original path. For point-to-point movements in free space (as in our example), this stopping strategy will remain on the planned path. Figure 3 shows the planned joint velocity and position, the stopping plan, and the joint velocity of a simulated robot.

(ii) **Deceleration to reduced speed.** When the signal arrives to slow down, a stopping trajectory is calculated as above.

[3] See the file ShortRangePreset.json in the wiki pages at [31].

[4] The complete setup including all experimental scenarios is illustrated in the accompanying video at `https://youtu.be/zP3c7Eq8yVk`.

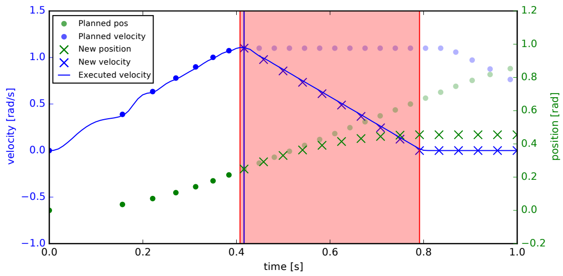


Fig. 3: Stopping motion using the trajectory controller. The stopping signal was received at 0.4077 s. The deceleration starts 0.0086 s later. The robot stops from the a speed of 1.1 rad /s in 0.3836 s (red area). The first red vertical line shows arrival of stop signal and the blue vertical line (0.0085 s later) marks the end of computation of the new trajectory. Note, that we consider the worst-case execution time in the selection of the reference state.

The original trajectory is scaled using the *IterativeParabolicTimeParameterization* (MoveIt!) to comply with the desired reduced speed. When the linear deceleration reaches the speed of the scaled trajectory, we search for the closest trajectory point ahead of the scaled trajectory. The scaled trajectory is shifted in time to continue after the deceleration and both trajectories are stitched at this point together. Acceleration back to full speed is performed similarly.

The target joint position commands were then passed to the KUKA Sunrise cabinet via the FRI interface.

We took a conservative approach in the design of our controller as follows: when “pps status” signaled a more restrictive regime, it was executed immediately; conversely, in the other direction, a filter was applied to warrant that the operator has left the area. The pipeline described above is not safety-rated and the high-level robot control is capable of performing a Stop Category 2 only.

### *E. Human keypoint 3D estimation and distance measurements*

The integral part of collision avoidance is to correctly estimate the position of the operator’s keypoints in space. We created a ROS node that processed data from the Realsense D435 camera using the Realsense Python API (2.17.1)[31] to collect aligned color and depth images. All our image operations also rely on OpenCV3[34].

The color images were sent to the OpenPose library Python API[35] to estimate human keypoints. For OpenPose, we use the COCO model and with the net resolution matching the input images. We also used the model’s confidence value to drop detections that were below 0.6 confidence as they were often false positives. This threshold was found by letting OpenPose analyze a scene without the human.

The resulting keypoint locations were then deprojected using the aligned depth image and thus we received the 3D coordinates of the operator in the camera’s frame of reference. These keypoints are represented as reference frames and added to the ROS transform library (called *tf*). The *tf* package stores the relationships between different coordinate frames in a tree structure, allowing for calculation of the position of the human keypoints w.r.t. the robot’s keypoints by using the relation between their frames.

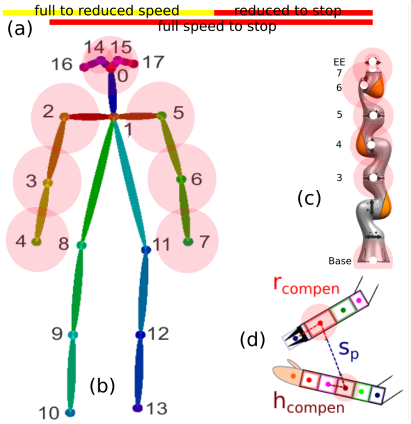


Fig. 4: Keypoints and bounding spheres representation (aspect ratio kept). (a) Stopping and stopping after reduced speed distances. (b) OpenPose keypoint distribution [6] with bounding spheres on the keypoints of interest. (c) KUKA LBR iiwa keypoints (picture source: KUKA LBR iiwa brochure) with compensation bounding spheres. (d) Schematic 2D separation distance calculation between robot and human keypoints. The compensation coefficients are the distances between the keypoints and the farthest point of the body that belongs to the body part near the keypoint.

Our experiment takes into account only upper body and hip keypoints detected by OpenPose’s posture model (see Fig. 4b), namely keypoints 0–7 and 14–17. These are the most relevant keypoints to our application and assume standard behavior of the operator. What we consider for our experiment as the human *head* are the keypoints of the nose (0), eyes (14, 15) and ears (16, 17).

### *F. Keypoint “bounding spheres”*

Discrete keypoints allow a faster calculation of distances and unambiguous interpretability of the system’s expected behavior. Nevertheless, they do not take into account the full occupancy of the bodies, which could lead to the underestimation of the real separation distance. This problem is especially relevant with sparsely placed keypoints.

We need to guarantee $S_{\mathrm{p}}$, the protective separation distance [2]. For this purpose, we introduce compensation coefficients for the robot $r_{\mathrm{compen}}$ and the human $h_{\mathrm{compen}}$.

The calculation of the compensation coefficients with given keypoints is divided into two steps. In the first step, every part of the body is assigned to its nearest keypoint. Then, for every keypoint, the maximal distance over all its assigned part (from the first step) is selected as the compensation coefficient (see Fig. 4d)—thereby guaranteeing the separation distance $S_\mathrm{p}$ in all cases. With increasing density of the keypoints, the compensation coefficients get smaller.

In our case, the robot compensation values were determined from the model of the robot. For the human, the values were assigned empirically based on the distribution of OpenPose keypoints (Table I). The human operator was interacting with the robot only with his upper body and the lower body was not taken into account. The resulting bounding spheres are in Fig. 4 and the values are in Table I.

| **EE** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **Base** |
|---|---|---|---|---|---|---|---|---|
| 0.01 | 0.11 | 0.15 | 0.15 | 0.15 | 0.15 | 0.15 | 0.14 | 0.10 |
| | **Nose** | **Neck** | **Eye** | **Ear** | **Arm** | **Elbow** | **Wrist** | |
| | 0.10 | 0.25 | 0.10 | 0.10 | 0.15 | 0.15 | 0.15 | |

TABLE I: Robot $\mathbf{r}_\mathrm{compen}$ and human $\mathbf{h}_\mathrm{compen}$ compensation values in meters.

### G. Protective separation distance

The *protective separation distance* is the “shortest permissible distance between any moving hazardous part of the robot system and any human in the collaborative workspace”, $S_p$, and it is described in [2] by the following formula:

$$S_p(t_0) = S_\mathrm{h} + S_\mathrm{r} + S_\mathrm{s} + C + Z_\mathrm{d} + Z_\mathrm{r} \tag{6}$$

with

$S_\mathrm{h}$ contribution to the $S_p(t_0)$ attributable to the operators change in location;

$S_\mathrm{r}$ contribution to the $S_p(t_0)$ attributable to the robot systems reaction time;

$S_\mathrm{s}$ contribution to the $S_p(t_0)$ due to the robot systems stopping distance;

$C$ distance that a part of the body can intrude into the sensing field before it is detected;

$Z_\mathrm{d}$ position uncertainty of the operator in the collaborative workspace, as measured by the presence sensing device resulting from the sensing system measurement tolerance;

$Z_\mathrm{r}$ position uncertainty of the robot system from the accuracy of the robot position measurement.

$S_p(t_0)$ can either be calculated dynamically or, as in our case, a fixed value based on worst case situation. Eq. 6 applies to all personnel in the collaborative workspace and to all moving parts of the robot system. In our case, we calculated the necessary stopping distance based on the maximal robot end-effector speed measured during the robot's unconstrained movement. The contributions marked as $S_i$ are determined using the robot's maximal speed $v_{max}$ multiplied with the appropriate $t_i$, so for example it should be $S_\mathrm{r} = t_\mathrm{r} \cdot v_{max}$. However, we used the average robot speed, $v_\mathrm{r} = \frac{v_\mathrm{max} - 0}{2}$, in our calculations in order to simulate the robot's slowing down during the stopping movement. This is a slight alteration of the very conservative demands of [2].

We determined the terms of Eq. 6 as follows:

$S_\mathrm{h}$ $(t_\mathrm{r} + t_\mathrm{s}) \cdot v_\mathrm{h}$, where $v_\mathrm{h}$ is the default human walking speed (1.6 m/s) [2], $t_\mathrm{r}$ is the time it took the robot to react to a issued stop status (0.1 s), and $t_\mathrm{s}$ the time it took the robot to stop its movement: 0.43 s, thus $1.6 \cdot (0.1 + 0.43) = 0.85$ m;

$S_\mathrm{r}$ $t_\mathrm{r} \cdot v_\mathrm{max}$ = $0.1 \cdot 1 = 0.1$ m;

$S_\mathrm{s}$ $t_\mathrm{s} \cdot v_\mathrm{r}$ = $0.43 \cdot 0.5 = 0.22$ m;

$C$ the setup did not allow the operator to enter the workspace without being detected: 0 m;

$Z_\mathrm{d}$ see the $h_\mathrm{compen}$ values from Subsection III-F: 0 m;

$Z_\mathrm{r}$ the LBR iiwa's repeatability value: 0.0001 m.

The time $t_\mathrm{s}$ was determined based on measured calculation times (0.005 s) and the maximal deceleration of the robot which was set to 1.5 rad /s$^2$.

Using these values, we can calculate the $S_p$ as in Eq. 7.

$$S_p(t_0) = 0.85 + 0.1 + 0.22 + 0.0001 = 1.17 \text{ m} \tag{7}$$

### H. Power and force limiting

The SSM regime prescribes that the robot stops before contact occurs. In our approach, we also allow the robot to slow down so that it can operate in the PFL regime, see below. We assume the end-effector exerts pressure on a surface area of at least 1 cm $^2$.

We can calculate the maximal relative speed of the system for a transient contact given the surface and the robot weight. For this, we use the formula A.6 from [2]. This equation also asks for some preliminary calculations, like for example $\mu$, the reduced mass for the two body system of the robot and the human operator. We summarize the calculation here. In order to ascertain absolute safety, we assume the worst case scenario, i.e. an impact in the chest. The values for $m_\mathrm{h}$, $p_\mathrm{max}$ and $k$ are taken from the appropriate tables in [2].

$$m_\mathrm{r} = \frac{M}{2} + m_L = \frac{23.9}{2} + 0 \tag{8}$$

$$\mu = \left(\frac{1}{m_\mathrm{h}} + \frac{1}{m_\mathrm{r}}\right)^{-1} = \left(\frac{1}{40} + \frac{2}{23.9}\right)^{-1} \tag{9}$$

$$v_{rel,max} = \frac{p_{max} \cdot A}{\sqrt{\mu \cdot K}} = \frac{2.4 \times 10^6 \cdot 1 \times 10^{-4}}{\sqrt{\mu \cdot 2.5 \times 10^4}} = 0.50 \tag{10}$$

Thus we know that the speed of 0.42 m/s is a conservative speed in order to be in the PFL regime. We determine the distance at which the robot needs to start slowing down to be PFL compliant in the same way as we did with SSM in Eq. 7. However, we take into account only the difference between 1 m/s and 0.42 m/s. The resulting value for $S_p$ is 0.73 m (full to reduced speed). The stopping distance for 0.42 m/s according to the equation would be 0.60 m (reduced to stop). According to [2], non-zero energy contact with the human head is not allowed. Thus our final setup forces the robot to stop on the proximity of the human head (see Section IV-C).

### I. Keypoint separation distance representation

The separation distance is represented in a matrix of minimal effective separation distances for every pair of human-robot keypoints that allow to meet the desired protective

| | Stop full | from speed | Reduce | speed | Stop reduced | from speed |
|---|---|---|---|---|---|---|
| | **Nose** | **Wrist** | **Nose** | **Wrist** | **Nose** | **Wrist** |
| **End-effector** | 1.28 | 1.33 | 1.44 | 1.49 | 0.71 | 0.76 |
| **3** | 1.33 | 1.38 | 1.49 | 1.54 | 0.76 | 0.81 |
| **Base** | 1.28 | 1.33 | 1.44 | 1.49 | 0.71 | 0.76 |

TABLE II: Effective keypoint-pair protective separation distance in meters.

separation distance for all. This matrix can be set explicitly or it can be a sum of different matrices as in our case.

The resulting separation distance is composed of several components—a $baseline$ and any terms relevant from the safety perspective. The $baseline$ is determined by the experimenter or calculated according to the methodology described together with Eq. 6 in Sec. III-G. We have to evaluate the maximum possible speed and the protective separation distance based on the "worst cases over the entire course of the application"[2]. The resulting keypoints $S^{ij}_{\mathrm{p}}$ are added to compensation coefficients based on the bounding spheres $\mathbf{h}_{\mathrm{compen}}$ and $\mathbf{r}_{\mathrm{compen}}$ described already in Sec.III-F.

This addition leads to the keypoint separation distances $S^{i,j}_{\mathrm{kp}}$ between any two given keypoints $i$, $j$.

$$S^{ij}_{\mathrm{kp}} = \; h^{i}_{\mathrm{compen}} + \; S^{ij}_{\mathrm{p}} \; + r^{j}_{\mathrm{compen}} \tag{11}$$

Thus we calculate the keypoint separation distances for each keypoint pair. We show two calculations: (1) According to SSM, the values necessary for a cat. 2 stop from full speed based on the Eq. 7 with the addition of the compensation values from Table I according to Eq. 11 are shown in Table II (left). (2) Combination of SSM and PFL regimes: robot first slows down and then stops only if needed. We add the calculations from Section III-G; the resulting values are in Table II (middle). An example is provided in Eq. 12 with the nose-end-effector keypoint pair. Reduced speed is triggered at the distance $S^{i,j}_{\mathrm{reduced,kp}}$ that is composed of $S_{\mathrm{fulltoreduced}}$ per PFL (Section III-H) and $S_{\mathrm{reducedtostop,kp}}$ per SSM (Section III-G, Table II, last column).

$$\begin{array}{cccccc} S^{i,j}_{\mathrm{reduced,kp}} & = & S_{\mathrm{fulltoreduced}} & + & S^{i,j}_{\mathrm{reducedtostop,kp}} \\ 1.44 & = & 0.73 & + & 0.71 \end{array} \tag{12}$$

Because of the shape of the KUKA robot, the values result in similar effective $S_{kp}$; accordingly we list three keypoints from the robot and omit duplicate keypoint-pair values.

## IV. Results

The robot performs a mock pick-and-place task; the operator periodically replaces one of the objects, entering the robot workspace. The robot responds appropriately by slowing down or stopping and resumes operation whenever possible. The scenarios contrast the standard approach of a zone scanner or safety mat (Sc. 1, 2) with the pairwise distance evaluation between operator and robot keypoints (Sc. 3-5). Some scenarios employ a safe reduced speed per PFL (Sc. 2, 4, 5) and Sc. 5 issues a stop only on human head proximity. The description of the scenarios in our implementation (Sec. IV-A – IV-C) is followed by a performance comparison on the mock task (Sec. IV-D). All upper body keypoints (see Fig. 4, right) were considered at all times, but we show only the safety-inducing keypoints in the plots below for clarity.

### *A. Scenario 1 and 2: Robot base vs. human keypoints*

In the first two scenarios, the distances between the robot base and the human keypoints were considered. The baseline $S_{\mathrm{p}}$ of 1.17 m (Eq. 7) is extended by compensation coefficients specific to the human keypoint bounding spheres (Sec. III-F, Table II). In addition, as only the base of the manipulator is considered, the robot's maximum reach of 0.8 m has to be added, giving 1.17+0.8 m, plus keypoint compensations.

In a similar manner, the second scenario approximated the setting with distance-based zones for reduced speed and stopping by using the values from Sec. III-H. A reduced speed zone started at 2.13 m (0.73+0.6+0.8) and stop at 1.40 m (0.6+0.8). The separation distance for slowing down from the maximum velocity was a composition of the necessary distance for slowing down, the necessary distance to stop from the reduced speed, and the robot's reach, see Fig. 4a.

### *B. Scenario 3 and 4: Robot vs. human keypoints*

In Scenario 3, we measure keypoint-pair separation distance with respect to the robot's moving parts (namely any joint above joint 3) to stop at $S_{\mathrm{p}}$ = 1.17 m. The fourth scenario involved a reduced speed zone (see Sec. III-H). When a human keypoint got closer than 1.33 m to any of the moving robot keypoints, the robot slowed down. If the human got closer than 0.60 m, the robot stopped. The behavior of the system is illustrated in Fig. 5.

### *C. Scenario 5: Addition of keypoint discrimination*

The last scenario described the case when the robot reacted with a stop only if the human head was closer than 0.60 m to the robot. Otherwise, the robot slows down (keypoint distance below 1.33 m). The behavior is illustrated in Fig. 6. Notice that the safety regimes of the robot were triggered by different keypoint pairs than in the case of the previous scenario in Fig. 5.

### *D. Performance in mock task*

Here we quantitatively evaluate the performance on the task under the different "safety regimes" as described above. The robot performs the task 20 times (measured at one of the two target objects) and the time needed is recorded. As a baseline, we use the unobstructed task at full speed of the robot and reduced speed. The full speed scenario would not comply with collaborative operation; reduced speed at all times would comply, provided the operator head is protected.

The results are shown in Table III. Operating the robot in the reduced speed PFL compliant regime, scenarios 4 and 5, outperformed most of the experimental scenarios. The scenarios that take pairwise distances between robot and operator keypoints into account and use two thresholds

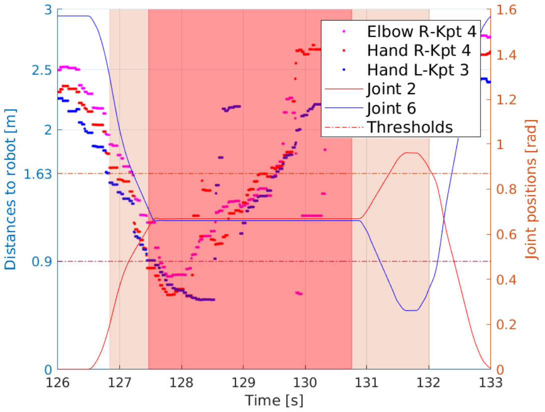


Fig. 5: Scenario 4: Reduced speed (light area) or stop (dark) triggered by keypoint distances below threshold. Positions of selected joints showing the slowing down / stopping (continuous lines, right y-axis). Keypoint pair distances triggering the behavior are shown (individual data points, left y-axis). Relevant threshold values: Reduced speed at 1.63 m and the stopping behavior at 0.90 m. These values are based on Eq. 12 and the appropriate compensation values from Table I.

(scenario 4 and 5) performed better than all other collaborative regimes. The last scenario that stops only for the head keypoints achieves the best performance.

| Full sp. | Reduced sp. | Sc. 1 | Sc. 2 | Sc. 3 | Sc. 4 | Sc. 5 |
|---|---|---|---|---|---|---|
| 154 | 256 | 267 | 254 | 257 | 231 | 228 |

TABLE III: Task duration for different scenarios in seconds.

## V. Discussion and conclusion

In this work, we used a robot in a mock collaborative scenario, in which it shares its workspace with a human. The operator's position was perceived with an Intel RealSense RGB-D sensor and human keypoints were extracted using OpenPose. Our paper presents an application of the standard for collaborative robot operation ISO/TS 15066 [2]. The standard prescribes two collaborative regimes (SSM and PFL). However, to our knowledge, there is no work considering both in a single application. We follow the standard to derive the protective separation distance (per SSM) and calculate the reduced robot velocity (in compliance with PFL constraints) and deploy them in a single framework. We demonstrate this union with an implementation of pairwise keypoint distance monitoring. Compared to classical zone monitoring, the keypoint distance method has higher resolution and constraints robot operation less. Also, keypoints can be treated differently, taking the sensitivity of human body parts or robot keypoints (e.g. sharpe edges) into account—in this way the constraints on collisions (per PFL) can be transformed into separation distances (per SSM).

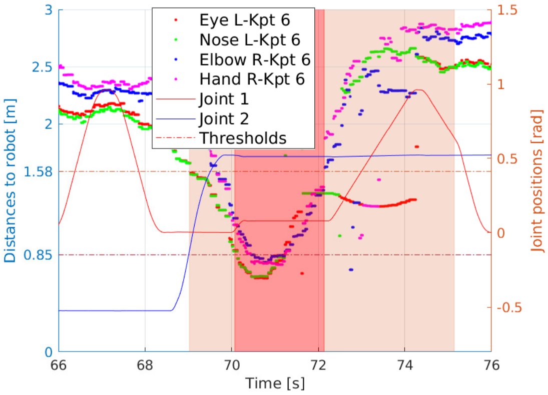


Fig. 6: Scenario 5. See also caption of Fig. 5. As soon as the first threshold at 1.58 m is met, the robot reacts with slowing down. When the human operator crosses the second threshold at 0.85 m with his head, the robot stops. Thresholds contain the compensation from Sec. III-F. Notice that the detection of the operator's elbow below the threshold does not trigger a stop but it does lead to a longer reduced speed period.

The operation of this framework was illustrated with a KUKA LBR iiwa robot interacting with a human partner that is perceived by a RGB-D sensor during a mock collaborative task. Contrasting a classical "stop zone" from the robot base with the keypoint-based approaches confirmed the potential of the distance monitoring between pairs of keypoints.

Multiple features could enhance our setup, notably we could add dynamic protective separation distances and occlusion compensation. The current approach monitors only positions and uses the maximum speeds for calculations. Instead, we could monitor relative speed and dynamically modify the protective separation distance accordingly.

Currently, occlusions could cause a misestimation of the human's keypoint location and thus the distance. Possible compensations and thus future enhancements are to use multiple sensors, compensate for occlusion by creating a human model or filter out the robot body in the scene. With these additions we could also incorporate active evasion of the human instead of our current reactive behavior (see [11]).

RGB-D sensors are not safety-rated yet. The reliability of the current sensors can be improved by combining multiple sensors and fusing the information from them [36], [37]. However, there is a clear need of safety-rated devices similar to those for zone monitoring that will provide 3D object coordinates and possibly human keypoint extraction: certified products are expected to appear on the market soon. The availability of such technology would dramatically expand the possibilities of human-robot collaboration in the SSM regime. Furthermore, as illustrated in this work, exploiting the "keypoint semantics" (e.g. chest vs. head) can be combined with the safety requirements as per PFL.

## ACKNOWLEDGMENT

This work was supported by the Czech Science Foundation, GA17-15697Y (P.S., M.H.); the Technological Agency of the Czech Republic, TJ01000470 (M.T.); the Czech Technical University in Prague, grant No. SGS18/138/OHK3/2T/13 (P.S.); the European Regional Development Fund, "Research Center for Informatics" (CZ.02.1.01/0.0/0.0/16_019/0000765) (P.S.); "Robotics for Industry 4.0" (CZ.02.1.01/0.0/0.0/15 003/0000470) (J.K.B.). We thank Karla Stepanova for assistance, Zdenek Straka for his previous work [30]. We are also indebted to Vasek Hlavac, Valentyn Cihala, Libor Wagner, Vladimir Petrik, Vladimir Smutny, and Pavel Krsek from CIIRC for their kind support in using the KUKA robot.

## REFERENCES

[1] "ISO 10218 Robots and robotic devices – Safety requirements for industrial robots," International Organization for Standardization, Geneva, CH, Standard, 2011.

[2] "ISO/TS 15066 Robots and robotic devices – Collaborative robots," International Organization for Standardization, Geneva, CH, Standard, 2016.

[3] J. Bessler, L. Schaake, C. Bidard, J. H. Buurke, A. E. B. Lassen, K. Nielsen, J. Saenz, and F. Vicentini, "Covr – towards simplified evaluation and validation of collaborative robotics applications across a wide range of domains based on robot safety skills," in *Wearable Robotics: Challenges and Trends*, M. C. Carrozza, S. Micera, and J. L. Pons, Eds. Cham: Springer International Publishing, 2019, pp. 123–126.

[4] S. Haddadin and E. Croft, "Physical human-robot interaction," in *Springer Handbook of Robotics*, 2nd ed., B. Siciliano and O. Khatib, Eds. Springer, 2016, pp. 1835–1874.

[5] S. Haddadin, A. De Luca, and A. Albu-Schäffer, "Robot collisions: A survey on detection, isolation, and identification," *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1292–1312, 2017.

[6] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *CVPR*, vol. 1, no. 2, 2017, p. 7.

[7] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele, "Deepercut: A deeper, stronger, and faster multi-person pose estimation model," in *European Conference on Computer Vision*. Springer, 2016, pp. 34–50.

[8] R. A. Güler, N. Neverova, and I. Kokkinos, "Densepose: Dense human pose estimation in the wild," *arXiv preprint arXiv:1802.00434*, 2018.

[9] S. Savazzi, V. Rampa, F. Vicentini, and M. Giussani, "Device-free human sensing and localization in collaborative human–robot workspaces: A case study," *IEEE Sensors Journal*, vol. 16, no. 5, pp. 1253–1264, 2016.

[10] F. Flacco, T. Kröger, A. De Luca, and O. Khatib, "A depth space approach to human-robot collision avoidance," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 338–345.

[11] F. Flacco, T. Kroeger, A. De Luca, and O. Khatib, "A depth space approach for evaluating distance to objects," *Journal of Intelligent & Robotic Systems*, vol. 80, p. 7, 2015.

[12] P. Jiménez, F. Thomas, and C. Torras, "3d collision detection: a survey," *Computers & Graphics*, vol. 25, no. 2, pp. 269–285, 2001.

[13] M. P. Polverini, A. M. Zanchettin, and P. Rocco, "A computationally efficient safety assessment for collaborative robotics applications," *Robotics and Computer-Integrated Manufacturing*, vol. 46, pp. 25–37, 2017.

[14] A. M. Zanchettin, N. M. Ceriani, P. Rocco, H. Ding, and B. Matthias, "Safety in human-robot collaborative manufacturing environments: Metrics and control," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 882–893, 2016.

[15] B. Lacevic and P. Rocco, "Kinetostatic danger field-a novel safety assessment for human-robot interaction," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE, 2010, pp. 2169–2174.

[16] V. Magnanimo, S. Walther, L. Tecchia, C. Natale, and T. Guhl, "Safeguarding a mobile manipulator using dynamic safety fields," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. IEEE, 2016, pp. 2972–2977.

[17] A. Roncone, M. Hoffmann, U. Pattacini, L. Fadiga, and G. Metta, "Peripersonal space and margin of safety around the body: learning tactile-visual associations in a humanoid robot with artificial skin," *PLoS ONE*, vol. 11, no. 10, p. e0163713, 2016.

[18] D. H. P. Nguyen, M. Hoffmann, A. Roncone, U. Pattacini, and G. Metta, "Compact real-time avoidance on a humanoid robot for human-robot interaction," in *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*. ACM, 2018, pp. 416–424.

[19] S. LaValle and J. Kuffner, "Randomized kinodynamic planning," *Int. Journal of Robotics Research*, vol. 20, no. 5, pp. 378–400, 2001.

[20] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *The international journal of robotics research*, vol. 5, no. 1, pp. 90–98, 1986.

[21] O. Brock and O. Khatib, "Elastic strips: A framework for motion generation in human environments," *The International Journal of Robotics Research*, vol. 21, no. 12, pp. 1031–1052, 2002.

[22] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *The International Journal of Robotics Research*, vol. 17, no. 7, pp. 760–772, 1998.

[23] R. Vatcha and J. Xiao, "Perceiving guaranteed continuously collision-free robot trajectories in an unknown and unpredictable environment," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 1433–1438.

[24] A. De Luca and F. Flacco, "Integrated control for phri: Collision avoidance, detection, reaction and collaboration," in *Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEEE RAS & EMBS International Conference on*. IEEE, 2012, pp. 288–295.

[25] J. A. Marvel, "Performance metrics of speed and separation monitoring in shared workspaces," *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 2, pp. 405–414, 2013.

[26] C. Sloth and H. G. Petersen, "Computation of safe path velocity for collaborative robots," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 6142–6148.

[27] M. Nico, M. Hamad, M. Becker, A. G. Marin, and S. Haddadin, "Safety map: A unified representation for biomechanics impact data and robot instantaneous dynamic properties," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1880–1887, 2018.

[28] A. Meguenani, V. Padois, and P. Bidaud, "Control of robots sharing their workspace with humans: an energetic approach to safety," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 4678–4684.

[29] R. Rossi, M. P. Polverini, A. M. Zanchettin, and P. Rocco, "A pre-collision control strategy for human-robot interaction based on dissipated energy in potential inelastic impacts," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 26–31.

[30] P. Svarny, Z. Straka, and M. Hoffmann, "Toward safe separation distance monitoring from RGB-D sensors in human-robot interaction," in *International PhD Conference on Safe and Social Robotics (SSR-2018)*, 2018, pp. 11–14.

[31] Intel, "librealsense," https://github.com/IntelRealSense/librealsense, 2018, version 2.17.1; Accessed: 2019-02-24.

[32] D. Coleman, I. Sucan, S. Chitta, and N. Correll, "Reducing the barrier to entry of complex robotic software: a MoveIt! case study," *arXiv preprint arXiv:1404.3785*, 2014.

[33] V. G. Rao and D. S. Bernstein, "Naive control of the double integrator," *IEEE Control Systems Magazine*, vol. 21, no. 5, pp. 86–97, Oct. 2001.

[34] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[35] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields," in *arXiv preprint arXiv:1812.08008*, 2018.

[36] F. Fabrizio and A. De Luca, "Real-time computation of distance to dynamic obstacles with multiple depth sensors," *IEEE Robotics and Automation Letters*, vol. 2, no. 1, pp. 56–63, 2017.

[37] M. Ragaglia, A. M. Zanchettin, and P. Rocco, "Trajectory generation algorithm for safe human-robot collaboration based on multiple depth sensor measurements," *Mechatronics*, 2018.

# Appendix M

# 3D Collision-Force-Map for Safe Human-Robot Collaboration

Authors' version of:

Svarny, P., Rozlivek, J., Ruslter, L., and Hoffmann, M. (2021). 3D Collision-Force-Map for Safe Human-Robot Collaboration. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3829-3835.

DOI: `https://doi.org/10.1109/ICRA48506.2021.9561845`
Youtube video: `https://youtu.be/4eHsbe4EuHU`

Author contributions: The contribution of M. Hoffmann was 25%.

# 3D Collision-Force-Map for Safe Human-Robot Collaboration

Petr Svarny, Jakub Rozlivek, Lukas Rustler, and Matej Hoffmann

***Abstract*— The need to guarantee safety of collaborative robots limits their performance, in particular, their speed and hence cycle time. The standard ISO/TS 15066 defines the Power and Force Limiting operation mode and prescribes force thresholds that a moving robot is allowed to exert on human body parts during impact, along with a simple formula to obtain maximum allowed speed of the robot in the whole workspace. In this work, we measure the forces exerted by two collaborative manipulators (UR10e and KUKA LBR iiwa) moving downward against an impact measuring device. First, we empirically show that the impact forces can vary by more than 100 percent within the robot workspace. The forces are negatively correlated with the distance from the robot base and the height in the workspace. Second, we present a data-driven model, 3D Collision-Force-Map, predicting impact forces from distance, height, and velocity and demonstrate that it can be trained on a limited number of data points. Third, we analyze the force evolution upon impact and find that clamping never occurs for the UR10e. We show that formulas relating robot mass, velocity, and impact forces from ISO/TS 15066 are insufficient—leading both to significant underestimation and overestimation and thus to unnecessarily long cycle times or even dangerous applications. We propose an empirical method that can be deployed to quickly determine the optimal speed and position where a task can be safely performed with maximum efficiency.**

## I. Introduction

*Physical Human-Robot Interaction* (pHRI) or *Human-Robot Collaboration* (HRC) (e.g., [1]) is a dynamically growing research field. At the same time, in industry, the expectations associated with collaborative robots (or co-bots; robots designed for direct interaction with humans [2]) are high, but their uptake has been somewhat held up by their performance limitations derived from strict safety constraints. Various safety standards, especially ISO 10218 [3] and ISO/TS 15066 [4] (TS 15066 for short), formulate these safety demands. These standards currently list four modes of collaboration. While all HRC is "continuous, purposeful interaction associated with potential or accidental physical events" [5], only the Power and Force Limiting (PFL) mode permits physical contact between the robot and the human when the robot is still autonomously moving, provided that the impact force, pressure, and energy stay within prescribed limits [4]. Nevertheless, the force thresholds derived from the TS 15066 for the PFL mode enforce low operational velocities, especially if there is a risk of clamping (see for example [6] or [7]). These constraints motivate current investigations geared to overcoming the limitations.

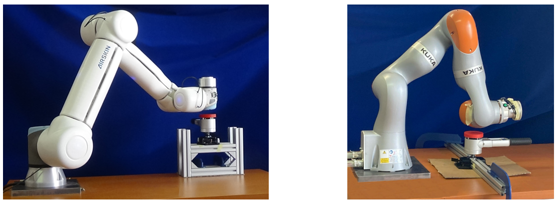

(a) UR10e. (b) KUKA LBR iiwa.

Fig. 1: Setup – robots and impact measuring device.

TS 15066 prescribes limits based on pain thresholds from studies like [8]. These limits are, however, subject to a heated debate, e.g. [6], [9], [10].

A detailed treatment of safety aspects of human-robot collisions is presented in [11], [12]. Contact modeling *per se* is notoriously difficult; in HRC, it is even more challenging as many parameters (mass and its distribution in colliding bodies, behavior of robot controller upon impact, etc.) are not known. Kovincic et al. [13], [14] suggest using collected impact data to model the impact forces using machine learning approaches because robot reaction mechanisms play a significant role in the resulting forces and are "not known or can not be identified" [13]. Schlotzhauer et al. [15] introduce a 2D Collision-Force-Map (2D CFM) and approximate the impact forces of UR10 and UR10e robots in a pick and place task with a second degree polynomial.

Finally, robot performance can be further boosted if PFL is not treated in isolation. For example, combinations with the Speed and Separation Monitoring collaboration mode—where robot needs to come to stop before contact—can be implemented (e.g., [16]). This can take the form of optimal velocity scaling [7], velocity scaling based on an impact force model [17], the use of control barrier functions [18], or by predicting the exerted force based on motor currents [19].

This work focuses on the PFL collaborative mode. We measure the forces exerted by two collaborative manipulators (UR10e and KUKA LBR iiwa) on an impact measuring device in different positions in the robot workspace and with various velocities (Fig. 1). Our approach is similar to 2D CFM [15] in that we use empirical measurements and fit a function relating robot position and speed to the impact force. Newly, we establish the importance of the height, leading to a 3D Collision-Force-Map (3D CFM).

**Our contributions are the following**: (i) a 3D collision-

This work was supported by the Czech Science Foundation (GA CR), project EXPRO (no. 20-24186X). P.S., J.R., and L.R. were additionally supported by the Czech Technical University in Prague, grant No. SGS20/128/OHK3/2T/13. The authors would like to thank Minh Thao Nguyenova, Martin Sramek, and Petr Posik for their valuable suggestions and Abdel-Nasser Sharkawy for assistance with the effective mass analysis.

All authors are with Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague. {petr.svarny, rozlijak, rustlluk, matej.hoffmann}@fel.cvut.cz

Digital Object Identifier (DOI): see top of this page.

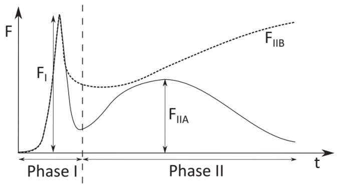


Fig. 2: Collision phases from [11]. Phase I, the initial dynamic impact with the force $\mathrm{F_I}$, and Phase II, either a diminishing force profile $\mathrm{F_{IIA}}$ in the case of no clamping or a non-diminishing force profile $\mathrm{F_{IIB}}$ if there is clamping.

force-map is created, considering the velocity, distance from robot base, and, newly, the height in the workspace; (ii) a simple data-driven model using only few samples is presented and validated; (iii) behavior of the two manipulators upon impact is analyzed, drawing important implications for their deployment in collaborative applications.

This article is structured as follows. The related theory is in Section II. The Experimental setup section is followed by Experiments and Results (IV). We close with Conclusion, Discussion, and Future work.

An accompanying video illustrating the experiments is available at `https://youtu.be/4eHsbe4EuHU`. The dataset with the data collected is at [20].

## II. Collision force maps and effective mass

### A. Power and Force Limiting

A human-robot collision can be decomposed into two phases (see [11]). An initial dynamic impact in Phase I is followed by the Phase II force profile that depends on the clamping nature of the incident (see Fig. 2). There are at least three possible scenarios (for details see [5]):

- unconstrained dynamic impact (no force in Phase II),
- constrained dynamic impact without clamping (diminishing force in Phase II)
- constrained dynamic impact with clamping (force is not diminishing in Phase II)

TS 15066 [4] does not make a distinction between the impact phases and merely distinguishes between two scenarios, a *transient* contact, i.e. dynamic impact that is unconstrained or is not followed by clamping, and *quasi-static* contact, i.e. dynamic impact followed by clamping. The equation A.6 from TS 15066 relating velocity ($v$) and (maximum) impact force ($F_{\max}$) is:

$$v \leq \frac{F_{\max}}{\sqrt{k}}\sqrt{m_R^{-1} + m_H^{-1}}, \tag{1}$$

with $m_R$ the effective robot mass, $m_H$ the human body part mass, $k$ the spring constant for the human body part and $F_{\max}$ the maximum impact force permitted for the given body region. As pointed out in [6], this is a simplified contact model with a single spring constant for the human body.

However, the risks cannot be evaluated solely based on the robot—it is necessary to take into account the application as a whole. We assume a mock pick and place scenario with a risk of a constrained dynamic impact on the human hand as in [15]. Contact may occur as the robot is descending towards the table, possibly clamping the hand of the operator. In practice, a risk analysis according to [21] will be required.

If we investigate constrained dynamic impacts, even without clamping, we can approximate $m_H^{-1} \approx 0$ as in [7][1]. This approximation allows us also to simplify the situation by investigating the relative velocity as simply the robot velocity with the human hand being still. The other variables are therefore set based on [4] as $F_{\max} = 140$ N and $k = 75000$ N/m. The moving masses of the UR and KUKA robot are approximately 30 kg and 20 kg respectively. Using the approximation from [4] that the effective robot mass $m_R$ is $M/2 + m_L$ (half of the total mass of the moving parts of the robot, plus the effective payload $m_L$, which is zero in our case), together with Eq. 1, would give permissible velocity up to 0.13 m/s for the UR robot and 0.16 m/s for the KUKA robot in case of clamping. If there is no clamping, the permissible force becomes 280 N and thus also the velocities are higher, namely 0.26 m/s for the UR and 0.32 m/s for the KUKA due to the weight difference between the robots.

### B. Collision-Force-Map – 2D and 3D

The assumptions and approximations made in [4] are too coarse and do not match empirical impact measurements. Schlotzhauer et al. [15] proposed a 2D Collision-Force-Map—a data-driven linear model to predict the impact force as a function of the distance from the z-axis of the robot base frame ($d$) and velocity ($v$). The model is a second degree polynomial of the form:

$$\ln(F) = \beta_0 + \beta_1 \cdot v + \beta_2 \cdot d + \beta_3 \cdot d^2 \tag{2}$$

The parameters are robot-, software-, and application-specific and should be found from a large number of measurements.

In this work, we add the height in the workspace ($h$) as an important additional dimension that affects the force exerted on impact. Euclidean distance in 3D between the end effector (EE) and the robot base would be a candidate representation, leading to a different 2D Collision-Force-Map. However, our empirical measurements—see Fig. 3—reveal a more complicated relationship between $d$, $h$, and $v$. For the UR10e robot (Left), the dependence of force on distance has a different profile for $h \geq 0.38$ m than for lower heights. This is true for two different speeds. For the KUKA (Right), the contribution of height to predicting the impact forces goes down with the distance from the base.

### C. Effective mass as function of d and h

We sought a theoretical rationale for the observations above. While TS 15066 considers the "effective robot mass" statically, Khatib [22] introduced the robot effective mass as a dynamic property depending on the robot's configuration and the impact direction. This has been later adopted by many others, sometimes also called reflected mass (e.g., [1],

[1] The impacted body part is constrained and thus immovable. Its weight in the PFL two-body spring model can be considered therefore as significantly larger than the other body's, and hence approximated as infinite.

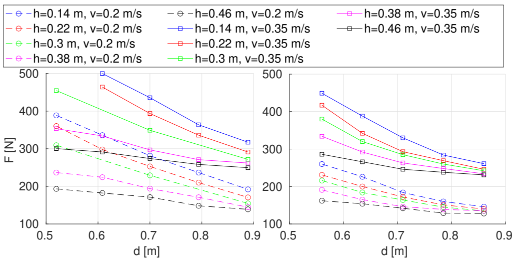
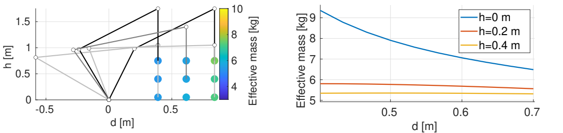


Fig. 3: Impact forces for different distances, heights, and velocities of EE. (Left) UR10e. (Right) KUKA LBR iiwa 7R800 with 10 Nm external torque limit.



(a) $m_R$ at 9 inspection points.

(b) $m_R$ as a function of $d$ and $h$.

Fig. 4: Calculating effective mass of model 3 DoF planar manipulator. Collision direction "down": $u = [0, -1]$.

[6], [7], [11], [23]). The effective mass of a manipulator in a given direction $\boldsymbol{u}$ can be modeled using the formula [23]:

$$m_{\boldsymbol{u}}^{-1} = \boldsymbol{u}^T [J(\boldsymbol{q}) M^{-1}(\boldsymbol{q}) J^T(\boldsymbol{q})] \boldsymbol{u}, \tag{3}$$

where $\boldsymbol{q}$ are the joint angles of a given position, $M(\boldsymbol{q})$ and $J(\boldsymbol{q})$ are the Inertia matrix and the Jacobian matrix of the manipulator, respectively (see, e.g., [24, Ch. 3 and Ch. 7]).

Although the robots have 6 (UR10e) and 7 (KUKA) degrees of freedom (DoF), the robot configurations at impact can be coarsely approximated with a 3 DoF planar manipulator. Inspired by the UR10e manipulator, we used a model with three links with masses $[13, 4, 4]$ in $kg$ and the length of the links $[0.5, 0.45, 0.05]$ in $m$. On a grid resembling Fig. 5, we used the analytical solution of inverse kinematics, restricted to the "elbow up" configuration, to reach with the EE the targets on the grid—see Fig. 4a—and calculated the effective mass, with $u = [0, -1]$ (collision in the downward direction). We sampled the workspace more densely, giving rise to Fig. 4b, providing a prediction in line with Fig. 3. The results also suggest that the effect of $d$ and $h$ should be considered together and "cross-factors" are needed.

### D. Acquiring 3D Collision-Force-Map from data

We investigated the significance of every element of the model like the one in Eq. 2, with additional terms in $h$ and terms with interaction factors between $d$, $h$, and $v$—for 3 datasets (Table I) simultaneously using a two-stage process. We started with the polynomial model containing all terms (variables $d$, $v$, $h$) up to degree three and their interaction terms up to degree three (19 terms together). We removed all terms with a $p$-value higher than 0.05 for all three datasets in stage one, to obtain 13 terms for stage two. In stage two, we iteratively removed terms and compared two model parameters: Root Mean Squared Error ($RMSE$) and coefficient of determination ($R^2$). In every iteration, the fit would typically be worse and hence $RMSE$ would increase and $R^2$ decrease. The term for which its removal produced the smallest change of these two parameters was removed. The change was defined as follows: $\sum_{\text{datasets}}(\Delta RMSE + 100\Delta R^2)$. The elimination procedure was stopped when this change for the term to be eliminated was bigger than 0.5 (i.e., removal of this term would make the fit significantly worse). The result of this process gave rise to Eq. 4:

$$\begin{aligned} \ln(F) = \beta_0 + \beta_1 \cdot v + \beta_2 \cdot d + \beta_3 \cdot d^2 + \beta_4 \cdot d \cdot h + \\ + \beta_5 \cdot h^2 + \beta_6 \cdot d^2 \cdot v + \beta_7 \cdot d \cdot v^2 + \beta_8 \cdot d \cdot h^2 \end{aligned} \tag{4}$$

## III. Experimental setup

### A. Setup and robots

An overview is in Fig. 1 and in the accompanying video. The experiments consisted of a series of impacts with the robots at different locations in the workspace and different speeds onto an impact measuring device. Both robots were commanded using the Cartesian linear movement—where the EE follows a straight line—toward the impact. As a large number of impacts were performed (more than 400 per robot in total), we preferred not to use the robot flange but the surface at the last joint instead. Robots were controlled using their standard control interfaces while experimental data were collected. We also specify the safety settings used for the experiments as they influence the robots' overall behavior and, in particular, the response to a collision.

*a) UR10e:* Our UR10e is equipped with additional protective layer, Airskin, that is not used in this work but the extra weight (1.8 kg) is considered. The worst-case data collection frequency was 800 Hz. The second most restrictive safety setting was used, which restricts the robot mainly in force and speed, but allows for sufficient acceleration and deceleration with respect to our velocities.[2]

*b) KUKA LBR iiwa 7R800:* The robot is equipped with a joint torque sensor in every axis and the safety setting restricts the maximal external torque at any joint. Two different settings were employed: 10 and 30 Nm. The data from the robot were collected with a frequency of 1000 Hz.

### B. Measuring device

We used *CBSF-75-Basic*—a measuring device (force transducer) for gauging forces and pressures, with a spring constant of 75000 N/m (see also [25]). Impact forces of up to 500 N can be measured, with maximum error up to 3 N (calibration protocol from supplier). Following TS 15066, appropriate damping material was added to mimic impacts on the back of the non-dominant hand. Peak force from the impact Phase I (Fig. 2) was recorded and used for analysis.

### C. Data collection

Schlotzhauer et al. [15] experimentally verified the rotational symmetry assumption. Thus, a single dimension, distance from the robot base, was the only relevant parameter.

[2] Allowed power: 200 W, Momentum: 10 kg m/s, Stopping time: 300 ms, Stopping distance: 0.3 m, Tool speed: 0.75 m/s, Tool force: 120 N, Elbow speed: 0.75 m/s, Elbow force: 120N.

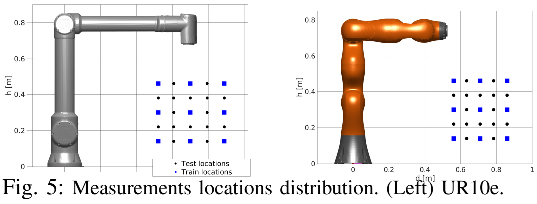


Fig. 5: Measurements locations distribution. (Left) UR10e. (Right) KUKA LBR iiwa 7R800.

| dataset | samples per state | training states (used samples*) | testing states (used samples*) |
|---|---|---|---|
| **UR10e** | 3 | 27 (75) | 88 (249) |
| **KUKA 30 Nm** | 3 | 27 (78) | 98 (291) |
| **KUKA 10 Nm** | 1 | 27 (26) | 98 (98) |

TABLE I: Collected datasets. A "state" is a combination of distance, height, and speed. *Samples that exceeded the measuring device limit of 500 N were not used.

In our case, it is sufficient to study a plane in the 3D workspace, varying two dimensions: $d$ and $h$.

For the UR robot, the $d$ ranged from 0.52 m to 0.89 m with increments of 0.09 m and five different heights from the level of the robot's base starting at 0.14 m with 0.08 m increments – see Fig. 5 (Left). The KUKA robot has a different reach. We sampled the workspace at the following positions: $d$ from 0.56 m to 0.86 m with an increment of 0.075 m and five heights corresponding to heights used with UR robot – see Fig. 5 (Right). At a given position, we performed measurements with five different velocities (0.20, 0.25, 0.30, 0.35, 0.40 m/s) in the downward direction.

*a) Training set:* It is our goal to develop a practical tool that can be rapidly deployed. Therefore, the number of measurements needed should be as small as possible. For training the model, we use only a subset of the grid—9 locations with blue square markers in Fig. 5—and 3 velocities (0.20, 0.30, 0.40 m/s). This gives rise to only 27 training measurements per robot, or 81 if every measurement is repeated 3 times. An overview is in Table I. For the KUKA robot, the repeatability was higher. Hence, for the 10 Nm ext. torque setting, measurements were performed only once.

*b) Testing set:* For every robot, 16 additional positions (black dots in Fig. 5) were tested with 5 velocities (0.20, 0.25, 0.30, 0.35, 0.40 m/s). For the 9 positions from the training set, only the velocities 0.25 and 0.35 m/s were added. In total, this gave 98 measurements per robot[3].

*c) Rotational symmetry verification:* In order to verify the assumption that rotation of the first joint does not influence the results, 117 additional measurements on the UR10e robot were performed.

All measurements above the recommended limit of the impact measuring device (500 N) were discarded.

## IV. Experiments and Results

Our results consist of a series of experiments in which two collaborative robots collide with an impact measuring device. An illustration is provided in the accompanying video. First, we present and evaluate the 3D Collision-Force-Map model for the two robots. Second, we compare the results obtained with the 2D CFM [15] and the treatment of Power and Force Limiting in TS 15066 [4]. Finally, we present the force profiles after impact and analyze their implications.

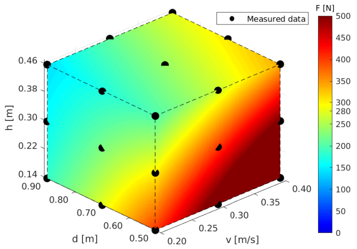


Fig. 6: UR10e – 4D visualization of 3D CFM model predictions within the sampled robot workspace for different collision velocities. The robot base is located at $d = 0$ and $h = 0$. Black points are showing measured training data capturing the measurement grid from Fig. 5 (Left).

### A. 3D Collision-Force-Map for UR10e

First, the rotational symmetry was experimentally verified using 117 measurements: 39 combinations of positions in the workspace and speed with 3 repetitions. The error, i.e. the difference in measured force on impact at different positions on the same circle (same $d$, $h$, and speed), was maximum 10 N (3.5 %), mean 1 N (0.05 %).

Second, restricting ourselves to a plane, we measured the impact forces on the grid of positions and at 5 different speeds (see Section III-C). Every measurement was repeated 3 times, with a maximum standard deviation (SD) of these three measurements of 3.85 N and a mean of these SDs across all locations/speeds of 1.12 N. In total, 324 measurements were performed.

The training set was used to fit the model of the form in Eq. 4. The obtained model was:

$$\begin{aligned} \ln(F) &= 6.2990 + 3.3761 \cdot v - 1.1050 \cdot d - \\ &-1.3066 \cdot d^2 - 1.5258 \cdot d \cdot h - 6.6954 \cdot h^2 + \\ &+4.0919 \cdot d^2 \cdot v - 6.0090 \cdot d \cdot v^2 + 8.5207 \cdot d \cdot h^2 \end{aligned} \tag{5}$$

Figure 6 shows the three variables, $h$, $d$, and $v$; only the surface of this color map is visible though. As would be expected, impact forces are directly proportional to the velocity. For a fixed $v$ and $h$ or $d$, a 2D visualization is possible—green lines in Fig. 7.

Table II quantifies the accuracy of our model on the testing set (we refer to the UR10e row here). We evaluate underestimation and overestimation of the impact forces separately—the former being more critical regarding safety assessment of the application. The maximal underestimation over the testing set (see Table I) is 3.56 % (8.42 N) and the mean underestimation is 1.37 % (4.45 N). Overestimation is higher, 6.35 % (22.45 N) at maximum and 2.40 % (6.97 N) on average. The error over the whole 3D CFM dataset is underestimating slightly more, 3.68 % (7.07 N). Overestimation

[3] Two positions (h = 0.3, d = 0.61; h = 0.3, d = 0.79) were omitted due to the experimenter's oversight on UR10e.

| dataset | | max UE<br>[% / N] | mean UE<br>[% / N] | max OE<br>[% / N] | mean OE<br>[% / N] |
|---|---|---|---|---|---|
| **UR10e** | **Ts** | 3.56 / 8.42 | 1.37 / 4.45 | 6.35 / 22.45 | 2.40 / 6.97 |
| | **All** | 3.68 / 7.07 | 1.50 / 4.63 | 6.35 / 22.45 | 2.16 / 6.26 |
| **Kuka 30 Nm** | **Ts** | 9.30 / 22.32 | 2.63 / 7.61 | 9.40 / 16.54 | 3.12 / 8.07 |
| | **All** | 9.30 / 22.32 | 2.58 / 7.37 | 9.40 / 16.54 | 3.08 / 7.93 |
| **Kuka 10 Nm** | **Ts** | 5.76 / 19.36 | 1.96 / 5.18 | 5.02 / 14.71 | 1.63 / 4.16 |
| | **All** | 5.93 / 20.20 | 1.94 / 5.26 | 5.38 / 19.48 | 1.59 / 4.11 |

TABLE II: Accuracy of 3D CFM model with underestimation (**UE**), overestimation (**OE**), the test set (**Ts**), complete dataset (**All**).

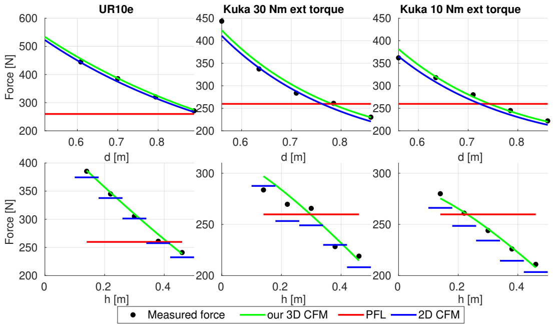


Fig. 7: Impact force model comparison of 3D CFM (ours) – green, 2D CFM [15] – blue, and the value for Power and Force Limiting mode from [4] – red. EE velocity = 0.30 m/s. (Top) Constant height of end effector in the workspace (0.14 m). (Bottom) Constant distance from base (0.70 m for UR10, 0.71 m for KUKA).

is more frequent for higher force values and underestimation for lower impact forces. The higher relative overestimation is probably due to the lower density of measurements for higher forces—when impact forces surpassed 500 N.

With a bound on the underestimation, the 3D CFM can be used to determine a safe speed for a collaborative application. Adding 10% to all predicted forces—a conservative choice—and knowing $d$, $h$, and allowed impact forces, one can rearrange Eq. 4 to obtain the maximum safe EE velocity.

### B. 3D Collision-Force-Map for KUKA LBR iiwa 7R800

Similarly to the UR robot, restricted to a plane, we measured impact forces on the grid of positions and at 5 different speeds (Section III-C, Fig. 5 (Right), Table I for details). We collected measurements with two different safety settings (30 and 10 Nm of max. external torque at any joint).

*a) 30 Nm external torque setting:* Every measurement, same location and speed, was repeated 3 times, with maximum SD of 3.09 N. The mean of these SDs across all locations/speeds was 0.58 N. The dataset is composed of 369 measurements (see Table I). The model, 3D CFM, for this robot and settings is given by the equation:

$$\begin{aligned}\ln(F) &= 7.0641 + 4.2943 \cdot v - 4.5286 \cdot d + \\ &+ 0.9917 \cdot d^2 - 0.5795 \cdot d \cdot h - 6.0074 \cdot h^2 + \\ &+ 3.9366 \cdot d^2 \cdot v - 7.2169 \cdot d \cdot v^2 + 7.0446 \cdot d \cdot h^2\end{aligned} \tag{6}$$

The results for one speed (0.30 m/s) are shown in Fig. 7, center, with the green line—with fixed height (0.14 m, top) or distance (0.71 m, bottom).

The accuracy of the model is quantified in Table II. The maximal underestimation error is 9.30 % (22.32 N) with a mean underestimation of 2.63 % (7.61 N) over the testing set and 2.58 % (7.37 N) over the whole 3D CFM dataset. The overestimation is comparable to the underestimation, with a mean value of 3.12 % (8.07 N) over the testing set and 3.08 % (7.93 N) over the whole 3D CFM dataset, and with the maximal error of 9.40 % (16.54 N). Both under- and overestimation are worse than in the case of the UR robot.

| dataset | | max UE<br>[% / N] | mean UE<br>[% / N] | max OE<br>[% / N] | mean OE<br>[% / N] |
|---|---|---|---|---|---|
| **UR10e** | **Ts** | 6.51 / 14.60 | 2.23 / 6.33 | 7.56 / 11.22 | 1.76 / 5.17 |
| | **All** | 6.51 / 14.60 | 2.27 / 6.42 | 9.13 / 13.21 | 2.78 / 7.98 |
| **KUKA 30 Nm** | **Ts** | 12.01 / 28.83 | 3.94 / 10.91 | 7.32 / 27.67 | 2.73 / 7.19 |
| | **All** | 12.01 / 28.83 | 3.75 / 10.37 | 7.89 / 22.14 | 2.84 / 7.46 |
| **KUKA 10 Nm** | **Ts** | 8.70 / 29.23 | 3.04 / 7.59 | 5.24 / 21.76 | 2.18 / 5.82 |
| | **All** | 8.70 / 29.23 | 3.00 / 7.53 | 5.24 / 21.76 | 2.12 / 5.43 |

TABLE III: Accuracy of 2D CFM models. Gray values indicate worse performance than 3D CFM model (Table II).

*b) 10 Nm external torque setting:* Due to the high repeatability of the measurements in the 30 Nm setting, deviations under the precision of the measurement device, only one measurement per position and speed was taken. The resulting dataset contains 124 measurements (see Table I). The resulting model, 3D CFM, for KUKA with 10 Nm is:

$$\begin{aligned}\ln(F) &= 6.6936 + 4.9297 \cdot v - 4.4782 \cdot d + \\ &+ 1.2926 \cdot d^2 - 0.3758 \cdot d \cdot h - 5.5669 \cdot h^2 + \\ &+ 3.2609 \cdot d^2 \cdot v - 7.2332 \cdot d \cdot v^2 + 6.4016 \cdot d \cdot h^2.\end{aligned} \tag{7}$$

The results for one speed (0.30 m/s) are shown in Fig. 7, right, with the green line—with fixed height (0.14 m, top) or distance (0.71 m, bottom). Compared to the 30 Nm setting, the forces are on average approximately 5% lower. The accuracy of the model is quantified in Table II. The maximal underestimation is lower than with the previous safety settings with 5.76 % (19.36 N) and an average of 1.96 % (5.18 N) over the testing set and 1.94 % (5.26 N) over the whole 3D CFM dataset. The overestimation is even lower than with the UR robot with a peak value of 5.02 % (14.71 N) and 1.63 % (4.16 N) on average over the testing set and 1.59 % (4.11 N) over the whole 3D CFM dataset.

### C. 3D Collision-Force Map vs. 2D CFM vs. PFL (TS 15066)

First, we want to compare our results with the 2D Collision-Force-Map (2D CFM) [15]. We used the least-squares method to train the 2D CFM model (Eq. 2) on our data, using the 0.20, 0.30, and 0.40 m/s EE velocities. A comparison for one velocity (0.30 m/s) and one height (0.14 m) is visualized in the top panels of Fig. 7. As the 2D CFM model does take $h$ into account and as we have shown the forces to importantly depend on this parameter, a single 2D CFM model will fail to deliver predictions on the whole workspace. To allow for a more fair comparison, we have trained it separately for every height—blue lines in the bottom panels of Fig. 7. As can be seen in Table III, the 2D CFM overestimation errors are comparable to our 3D CFM model errors (higher for UR and lower for KUKA 30 Nm dataset). On the other hand, the 2D CFM models underestimate significantly more than our 3D CFM model.

Power and Force Limiting according to [4] does not take $d$ or $h$ into account and considers velocity only (see Section II-

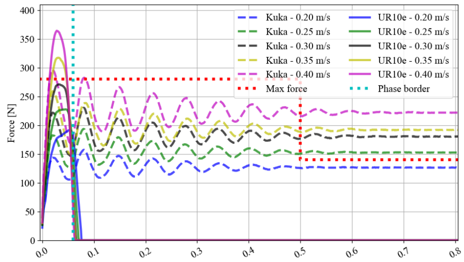


Fig. 8: Force evolution after impact. UR10e (at $d = 0.89$ m, $h = 0.14$ m) – solid lines. KUKA (at $d = 0.86$ m, $h = 0.14$ m, $30$ Nm ext. torque limit) – dashed lines. Phase I / Phase impact phase boundary (cf. Fig 2) – cyan dotted line. Permissible force per TS 15066 – red dotted line.

A). Eq. 1 can be rearranged and $F$ obtained. With the corresponding robot masses and $v = 0.3$, this gives rise to the red lines in Fig. 7. Clearly, such an approximation is insufficient. Moreover, next to overestimation, it leads also to gross underestimation of the impact forces and hence violates the safety of the human (by the very standards of [4]).

### D. Nature of dynamic impact

Peak force estimation is only one component required to assess safety of a HRC application. Collision force evolution after "Phase I" (Section II-A, Fig. 2) is also important. Fig. 8 shows this for a selection of our experiments. For the UR10e robot, only Phase I is present. That is, although the scenario has a "clamping nature", the UR10e controller makes the EE actively bounce back and thus makes the actual contact of a transient kind. On the other hand, the KUKA robot shows a prolonged damped harmonic movement upon impact.

TS 15066 prescribes maximum force thresholds for the first 0.5 s of impact (transient contact) and half this threshold afterward (quasi-static contact)—as shown in Fig. 8 with red dotted lines. Thus, based on our empirical findings, one could apply the higher force thresholds for the UR10e (e.g., 280 N) and only half that threshold for the KUKA (140 N), which would dramatically alter the safe speeds in the application.

## V. Conclusion, Discussion, Future Work

Using two collaborative robots, UR10e and KUKA LBR iiwa 7R800, we performed 934 measurements of forces exerted on the impact of the robot end-effector with an impact measuring device, with different robot velocities and at different locations in the robot workspace. The collision direction was always down, perpendicular to the table surface. We established a clear relationship between the distance from the robot base and the impact forces (in line with [15]) and, newly, also the height in the workspace—both variables being inversely proportional to the impact forces. We developed a data-driven model—3D Collision-Force-Map—that estimates the forces as a function of distance, height, and velocity, including their mutual relationships. This model is more accurate than 2D CFM [15] and PFL according to [4] that does not take position in the workspace into account. Furthermore, we show that it can be trained from a limited amount of data: we sampled only 9 positions in the workspace and 3 velocities to train the model.

Thus, our main contribution is a tool that allows for rapid prototyping of a collaborative robot workspace. For quasi-static impacts on the back of the hand, a force limit of 140 N is prescribed by TS 15066 [4], which would based on the formula from TS 15066 limit the allowed EE speed in the whole workspace to 0.13 and 0.16 m/s for the UR10e and KUKA LBR iiwa, respectively. Our measurements reveal that if the task is performed, for example, 0.8 m away and 0.4 m above the robot base, speeds of 0.16 m/s (UR10e) and 0.20 m/s (KUKA LBR iiwa) stay within the prescribed force limit. Furthermore, we observe that despite the clamping nature of our scenario, the UR10e robot generates only transient contact. With the 280 N limit, 0.36 m/s will still be safe with the UR10e—an almost threefold increase. The PFL formulas from TS 15066 are insufficient—leading both to significant underestimation and overestimation at different locations in the robot workspace, and thus to unnecessarily long cycle times or even dangerous applications. The impact measuring device was firmly attached to the table. The possibility that a human operator would be moving against the robot prior to collision was thus not considered. However, we focused on quasi-static contacts where the limits are stricter. The most dangerous part of the incident is in the clamping nature.

Interestingly, the trend of the relationship between distance from / height above the robot base and the forces exerted on collision is largely consistent across two different collaborative robots and also in line with our simple 3 DoF model. However, whether the effective mass entirely determines the trends in empirically measured forces remains an open question. In addition, the impacts are not uncontrolled. The collision is detected by the robot internal controllers, generating a response, which is likely quick enough to shape the force evolution even during the first phase of the impact. Extending the effective mass models is thus impeded by the fact that accurate inertial parameters of the manipulators are not known and the controllers are proprietary. In our case, different safety settings (external torque) resulted in different impact forces. Thus, empirical assessment of impact forces in the robot workspace seems indispensable at the moment.

It should be noted that our results are not expected to generalize to other robots or even different collision sites, directions, or kinematic configurations on the same manipulators. We concentrated on downward movement of the robot to the table, which is typical of many applications, and quasi-static contact, which represents the worst-case scenario. Impacts were made with the last robot joint, not the flange, for practical reasons. We propose an empirical method that can be deployed by robot integrators on a specific application site to quickly determine the optimal speed and position in the workspace where a task can be safely performed with maximum efficiency. The contact type and location on the robot and position in the workspace should all be set according to the application. In summary, for effective design of a collaborative application, empirical measurements are indispensable.

## References

[1] S. Haddadin and E. Croft, "Physical Human-Robot Interaction," in *Springer Handbook of Robotics*. Springer, 2016, pp. 1835–1874.

[2] "ISO 8373 Robots and robotic devices – Vocabulary," International Organization for Standardization, Geneva, CH, Tech. Rep., 2012.

[3] "ISO 10218 Robots and robotic devices – Safety requirements for industrial robots," ISO, Geneva, CH, Tech. Rep., 2011.

[4] "ISO/TS 15066 Robots and robotic devices – Collaborative robots," ISO, Geneva, CH, Tech. Rep., 2016.

[5] F. Vicentini, "Collaborative robotics: a survey," *Journal of Mechanical Design*, pp. 1–29, 2020.

[6] N. Mansfeld, M. Hamad, M. Becker, A. G. Marin, and S. Haddadin, "Safety Map: A Unified Representation for Biomechanics Impact Data and Robot Instantaneous Dynamic Properties," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1880–1887, 2018.

[7] N. Lucci, B. Lacevic, A. M. Zanchettin, and P. Rocco, "Combining Speed and Separation Monitoring with Power and Force Limiting for Safe Collaborative Robotics Applications," *IEEE Robotics and Automation Letters*, 2020.

[8] K. Suita, Y. Yamada, N. Tsuchida, K. Imai, H. Ikeda, and N. Sugimoto, "A failure-to-safety" Kyozon" system with simple contact detection and stop capabilities for safe human-autonomous robot coexistence," in *Robotics and Automation, International Conference on*, vol. 3. IEEE, 1995, pp. 3089–3096.

[9] D. Han, M. Y. Park, H. Shin, K. S. Kim, and S. Rhim, "Identifying Safety Conditions of Human-Robot Collision based on Skin Injury Analysis," in *International Conference on Ubiquitous Robots (UR)*. IEEE, 2018, pp. 420–423.

[10] M. Y. Park, D. Han, J. H. Lim, M. K. Shin, Y. R. Han, D. H. Kim, S. Rhim, and K. S. Kim, "Assessment of pressure pain thresholds in collisions with collaborative robots," *PLoS ONE*, vol. 14, no. 5, 2019.

[11] S. Haddadin, "Physical safety in robotics," in *Formal modeling and verification of cyber-physical systems*. Springer, 2015, pp. 249–271.

[12] S. Haddadin, A. De Luca, and A. Albu-Schäffer, "Robot collisions: A survey on detection, isolation, and identification," *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1292–1312, 2017.

[13] N. Kovincic, H. Gattringer, A. Müller, M. Weyrer, A. Schlotzhauer, L. Kaiser, and M. Brandstötter, "A model-based strategy for safety assessment of a robot arm interacting with humans," *PAMM*, vol. 19, no. 1, p. e201900247, 2019.

[14] N. Kovincic, H. Gattringer, M. Andreas, and M. Brandstötter, "A Boosted Decision Tree Approach for a Safe Human-Robot Collaboration in Quasi-static Impact Situations," *Advances in Service and Industrial Robotics: Results of RAAD*, p. 235, 2020.

[15] A. Schlotzhauer, L. Kaiser, J. Wachter, M. Brandstötter, and M. Hofbaur, "On the trustability of the safety measures of collaborative robots: 2D Collision-force-map of a sensitive manipulator for safe HRC," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2019, pp. 1676–1683.

[16] P. Svarny, M. Tesar, J. K. Behrens, and M. Hoffmann, "Safe physical HRI: Toward a unifed treatment of speed and separation monitoring together with power and force limiting," in *2019 IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, 2019, pp. 7580–7587.

[17] H. Shin, K. Seo, and S. Rhim, "Allowable maximum safe velocity control based on human-robot distance for collaborative robot," in *Int. Conf. on Ubiquitous Robots (UR)*. IEEE, 2018, pp. 401–405.

[18] F. Ferraguti, M. Bertuletti, C. T. Landi, M. Bonfe, C. Fantuzzi, and C. Secchi, "A control barrier function approach for maximizing performance while fulfilling to ISO/TS 15066 regulations," *IEEE Robotics and Automation Letters*, 2020.

[19] P. Aivaliotis, S. Aivaliotis, C. Gkournelos, K. Kokkalis, G. Michalos, and S. Makris, "Power and force limiting on industrial robots for human-robot collaboration," *Robotics and Computer-Integrated Manufacturing*, vol. 59, pp. 346–360, 2019.

[20] P. Svarny, J. Rozlivek, L. Rustler, and M. Hoffmann, "3D Collision Force Map – Public dataset," 2020. [Online]. Available: https://bit.ly/34D27B8

[21] "ISO 12100 Safety of machinery – General principles for design – Risk assessment and risk reduction," Tech. Rep.

[22] O. Khatib, "Inertial properties in robotic manipulation: An object-level framework," *The International Journal of Robotics Research*, vol. 14, no. 1, pp. 19–36, 1995.

[23] S.-D. Lee, B.-S. Kim, and J.-B. Song, "Human–robot collision model with effective mass and manipulability for design of a spatial manipulator," *Advanced Robotics*, vol. 27, no. 3, pp. 189–198, 2013.

[24] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: modelling, planning and control*. Springer Science & Business Media, 2010.

[25] D. Mewes and F. Mauser, "Safeguarding crushing points by limitation of forces," *International Journal of Occupational Safety and Ergonomics*, vol. 9, no. 2, pp. 177–191, 2003.