CTU

CZECH TECHNICAL UNIVERSITY IN PRAGUE

Faculty of Electrical Engineering
Department of Cybernetics

Programme: Electrical Engineering and Information Technology

Specialization: Artificial Intelligence and Biocybernetics

Doctoral thesis

Learning peripersonal space representations: From brains to robots

Zdeněk Straka

Supervisor specialist: doc. Mgr. Matěj Hoffmann, Ph.D.

Supervisor: prof. Ing. Tomáš Svoboda, Ph.D.

Prague, 2023

Acknowledgments

First of all, I would like to express my gratitude to my supervisors, Matej Hoffmann and Tomas Svoboda, for their guidance, support, time and patience. I would also like to thank them for many inspiring discussions and their friendly and empathetic attitude.

I also want to thank my colleagues, especially Petr Svarny, Filipe Gama, Shubhan Patni, Jason Khoury, Jakub Rozlivek, Lukas Rustler, Sergiu Popescu, Valentin Marcel, Hagen Lehmann, Karla Stepanova, Petr Posik, Jana Kostliva, Petra Ivanicova and Kristina Lukesova for their support and help.

I would like to express my appreciation to my entire family, particularly my mother and brother Radim, and my friends for their unwavering support during my academic journey. A special thank you goes out to my biggest fans of my academic journey, my grandma and my grandpa, who recently passed away, for their immeasurable love and support.

During my PhD studies, I have been supported by the following grants and projects:

- Czech Science Foundation (GA CR) (project No. 17-15697Y),
- Czech Science Foundation (GA CR) (project No. 20-24186X),
- European Regional Development Fund, "Research Center for Informatics" (project No. CZ.02.1.01/0.0/0.0/16_019/0000765),
- Grant Agency of the Czech Technical University in Prague (grant No. SGS16/161/OHK3/2T/13),
- Grant Agency of the Czech Technical University in Prague (grant No. SGS18/138/OHK3/2T/13),
- Grant Agency of the Czech Technical University in Prague (grant No. SGS20/128/OHK3/2T/13).

Contents

Abstract

The encoding of space in close proximity to the body, referred to as peripersonal space (PPS), is believed to contribute significantly to defensive behavior and consequently, survival. Despite numerous empirical findings on PPS representations in both humans and monkeys, the neural mechanisms underlying these processes remain largely unknown. In this thesis, we primarily investigate the idea that visuo-tactile (or audio-tactile) prediction—the use of a visual (or auditory) stimulus to predict a future contact with the body perceived through touch—is a key mechanism of PPS encoding, as initial evidence suggests. We investigated the mechanism using computational models. We started with the tactile modality and developed biologically-inspired representation of large areas of the body surface of the humanoid robot. The representation was based on a modified self-organizing map neural network, trained on tactile data from the humanoid robot, which allowed to incorporate prior knowledge about the representation of the body surface in the primary somatosensory cortex. This research was employed to investigate the process of formation of the representation. This was followed by models performing visuo-tactile prediction. One model employed a neural network architecture which combined a Restricted Boltzmann Machine to integrate position and velocity, with a feedforward neural network for predicting future contact with the body. The model demonstrated the feasibility of visuo-tactile prediction by replicating the phenomenon of PPS expansion, which occurs as velocity of an approaching stimulus increases. The second model was a Bayesian Decision Theory based normative model. This model successfully replicated many crucial characteristics of PPS encoding. In order to investigate the development of PPS encoding with child-caregiver-like interaction data, we created a model utilizing raw visuo-tactile inputs. We began with a deep neural network based on predictive coding for next-frame video prediction that achieved state-of-the-art performance. Subsequently, this network was extended to include the tactile modality and utilized raw visuo-tactile inputs generated during an interaction between a humanoid robot and a person, emulating the interaction between a child and a caregiver. The ongoing work report with preliminary results is included in the thesis.

Abstrakt

Předpokládá se, že reprezentace prostoru v těsné blízkosti těla (peripersonální prostor – PPP), významně přispívá k obrannému chování a tím i k přežití. Navzdory četným empirickým poznatkům o reprezentaci PPP u lidí i opic, zůstávají neurální mechanismy, které jsou základem těchto procesů, z velké části neznámé. V této práci se zabýváme především myšlenkou, že vizuálně-taktilní (nebo zvukově-taktilní) predikce – vizuální (nebo zvukový) podnět se využije k předpovědi budoucího kontaktu s tělem vnímaného prostřednictvím dotyku – je klíčovým mechanismem reprezentace PPP, jak naznačují první poznatky. Tento mechanismus jsme zkoumali pomocí výpočetních modelů. Začali jsme s hmatovou modalitou a vyvinuli jsme biologicky inspirovanou reprezentaci velkých ploch povrchu těla humanoidního robota. Reprezentace byla založena na modifikované neuronové síti samoorganizující se mapy, natrénované na taktilních datech z humanoidního robota, což umožnilo zahrnout předchozí znalosti o reprezentaci povrchu těla v primární somatosenzorické kůře. Tento výzkum byl využit ke zkoumání procesu tvorby této reprezentace. Následovaly modely provádějící vizuálně-taktilní predikci. První model využíval architekturu neuronové sítě, která kombinovala omezený Boltzmannův stroj pro integraci polohy a rychlosti s dopřednou neuronovou sítí pro předpovídání budoucího kontaktu s tělem. Model prokázal proveditelnost vizuálně-taktilní predikce replikováním jevu expanze PPP, ke kterému dochází s rostoucí rychlostí blížícího se podnětu. Druhým modelem byl normativní model založený na bayesovské teorii rozhodování. Tento model úspěšně replikoval mnoho klíčových charakteristik kódování PPP. Abychom prozkoumali vývoj reprezentace PPP s daty interakce podobné interakci dítěte a pečovatele, vytvořili jsme model využívající nezpracované vizuálně-taktilní vstupy. Začali jsme s hlubokou neuronovou sítí založenou na prediktivním kódování pro predikci dalšího snímku videa, která dosáhla výkonu na úrovni nejlepších aktuálních metod. Následně byla tato síť rozšířena o taktilní modalitu a využila nezpracované vizuálně-taktilní vstupy generované během interakce mezi humanoidním robotem a osobou, která napodobuje interakci mezi dítětem a pečovatelem. Zpráva o této probíhající práci s předběžnými výsledky je součástí práce.

Chapter 1

Introduction

Understanding how a complex biological system like the brain, with billions of interconnected neurons, works is one of the most challenging tasks for modern science. Due to the brain's unimaginable complexity, high non-linearity, and limited observability, solely relying on traditional experimental approaches is limiting for a thorough understanding.

In recent decades, however, advances in computer science have opened up new horizons for unraveling the mysteries of the brain. The new computational methods, such as machine learning techniques, have the potential to become powerful complements to the empirical approach. By bridging the gap between the theoretical and empirical approaches, the methods provide a valuable framework for deepening our insight into the mechanisms of different brain functions and behavior. *Abstract and often vague ideas of the potential mechanism can be transformed into computational models and compared with empirically observed properties.* The models which conflict with the empirical data are eliminated [1]. The process of using the computational models in cognitive neuroscience is shown in Fig. 1.1 (A). In this thesis, the models are primarily used to explore the biological mechanisms of encoding space in close proximity to the body and related events in that space.

The computational models of different phenomena generally strongly depend on available technology and computational methods. For example, without effective optimization methods and fast graphics processing units, it would not be possible to get state-of-the-art models of representations of images in inferior temporal cortex in monkeys and humans which are based on convolutional neural networks [3–5]. This dependence of the models on suitable methods is reflected in the second objective of the thesis.

In light of the opportunities which the intersection of computer science, engineering, cognitive neuroscience and psychology provides, there are three objectives of the thesis (see Fig. 1.1 (B)):

1. Building on the schema (see Fig. 1.1 (A)), the first objective of the thesis is to create computational models which deepen understanding of the biological mechanisms of the representations of the events within the close surroundings of the body.

2. The second objective is to provide new computational methods, preferably inspired by mechanisms observed in the brain, for the computational models. For example, we created a neural network for next frame video prediction based on predictive coding schema which was then used as a base of a visuo-tactile predictive model of peripersonal space encoding (see Chapter 3).

3. The third objective is to apply the computational models of PPS encoding and the computational methods in machine learning and robotics.

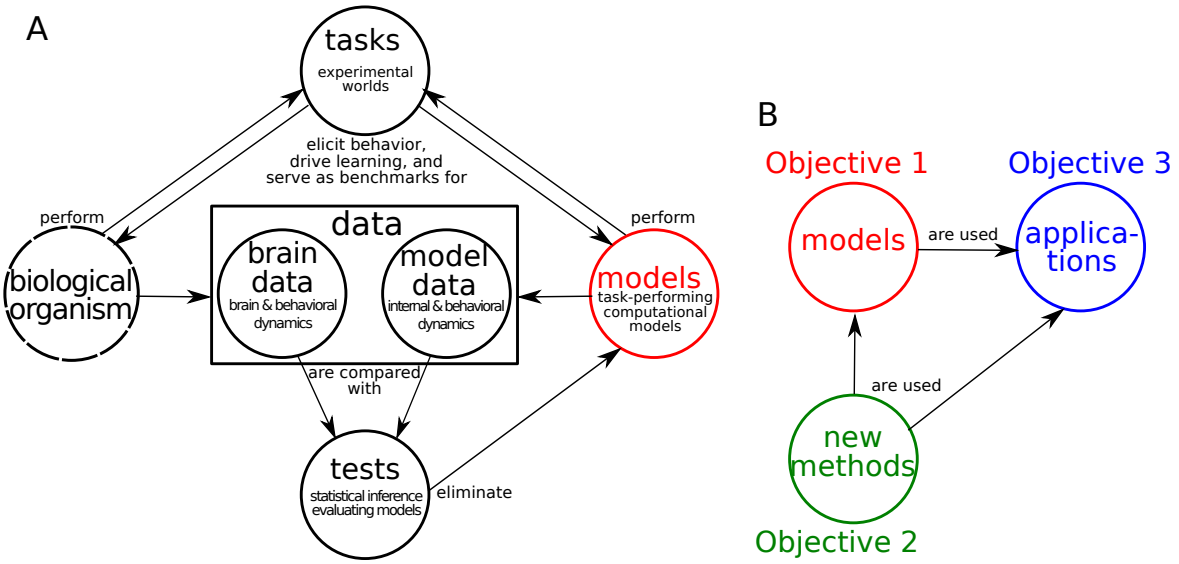

Figure 1.1: **(A) Schema of usage computational models in brain sciences from [1].** The biological organism and model perform an experimental task. The data from the organism and the model are collected and compared. If the brain and model data does not correspond, the model is eliminated. The models can also perform experimental tasks which may be performed with biological organisms in the future. For example, in case of PPS research the task can be observing of the approaching visual object by a monkey (e.g., [2]) and by a predictive PPS encoding model. In this case, firing rate of a PPS-related neuron is recorded and compared with predicted tactile activations by the model. Schema redrawn from [1]. **(B) Objectives of the thesis. Objective 1** — create computational models which deepen understanding of the biological mechanisms of the representations of the events within the close surroundings of the body. **Objective 2** — provide new computational methods, preferably inspired by mechanisms observed in the brain, for the models. **Objective 3** — apply the computational models of PPS encoding and the computational methods in machine learning and robotics. See text for details.

Individual contribution of each work to these objectives is elaborated in Chapter 2 (see Table 2.1 for summary).

In Fig. 1.2, the objectives of the thesis are elaborated from the perspective of synthetic methodology (SM), also known as "understanding by building" [6] (Chapter 1). A synthetic model of natural phenomenon is created. The created model is used for deepening of the understanding of the natural phenomenon. The model may be eventually transformed into application.

1.1 Peripersonal space representation as a prediction task

The central topic of the thesis is peripersonal space (PPS), which is the space in the close surroundings of the body [7,8] (see Fig. 1.3), and how it is together with corresponding events (e.g., presence of a predator) represented by the brain. It has special importance for survival, because objects in the PPS may come into direct contact with the body. Experimental findings indicate that PPS encoding plays a significant role in defensive behavior (see e.g., [9]). When a flying stone or another dangerous object approaches the body, it can be a matter of life and death whether a defensive response is initiated. It is therefore not surprising that there

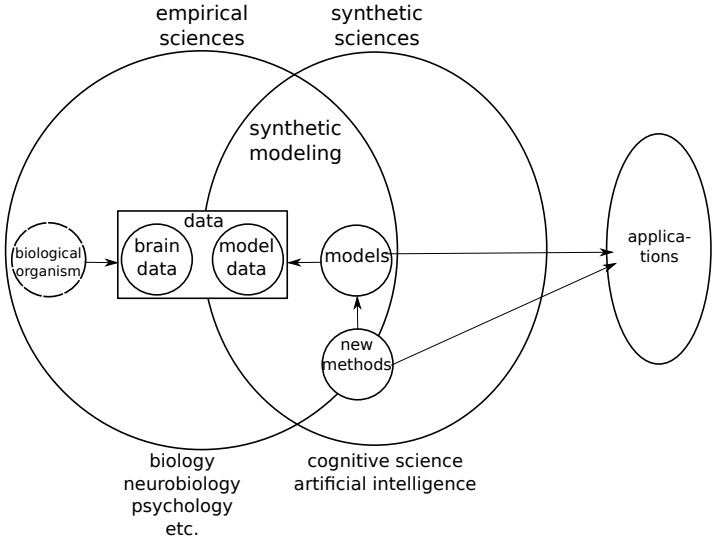

Figure 1.2: **Synthetic methodology and thesis objectives.** Overview of approaches to the study of intelligence. On the left, we have the empirical sciences like biology, neurobiology, and psychology that mostly follow an analytical approach. In the center, we have the synthetic ones, namely cognitive science and AI, which can either model natural agents (this is called synthetic modeling) 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 (this part of caption taken with a minor modification from [6], Chapter 1). Selected parts of the schema of usage computational models in brain sciences and thesis objectives—"models", "new methods" and "applications"—(see Fig. 1.1) were put into the synthetic methodology schema. Synthetic methodology schema redrawn with modifications from [6] (Chapter 1).

is a specialized circuit in the fronto-parietal cerebral cortex that responds to events in the immediate surroundings of the body (see e.g., [11, 12]).

Despite the existence of many empirical results, understanding of the mechanisms behind the PPS representations is still missing. Results of several empirical studies suggest that PPS encoding has an important role in impact prediction (e.g., [13,14]). This role of PPS encoding for impact prediction is also widely popular in the community (see [15,16] for reviews).

The idea of PPS encoding as impact prediction assumes that a visual or auditory stimulus is used to predict a future tactile stimulus. The predicted tactile impact value may also reflect a more general criterion than simply minimizing the difference between the predicted value and the actual tactile impact value. For example, it may be reflected that the cost of not predicting an impact when it occurs may be greater than the cost of predicting an impact when it does not occur (as we model in [17]). In the thesis, this concept will be referred to as *predictive PPS encoding*.

Motivated by the need to explore the idea of predictive PPS encoding in a more profound and specific way—from a vague idea to a model with behavior which can be compared to empirical results—, we mainly focused on the models which perform impact prediction. This choice was supported by the fact that it can be formulated as a computational task, which has moreover its significance in machine learning and robotics applications. As explained earlier (see Fig. 1.1 for a corresponding schema) comparing the behavior and properties of the model with empirical results may result in a hypothetical mechanism being rejected. The comparison should not be only with the current empirical results. It is highly desirable to

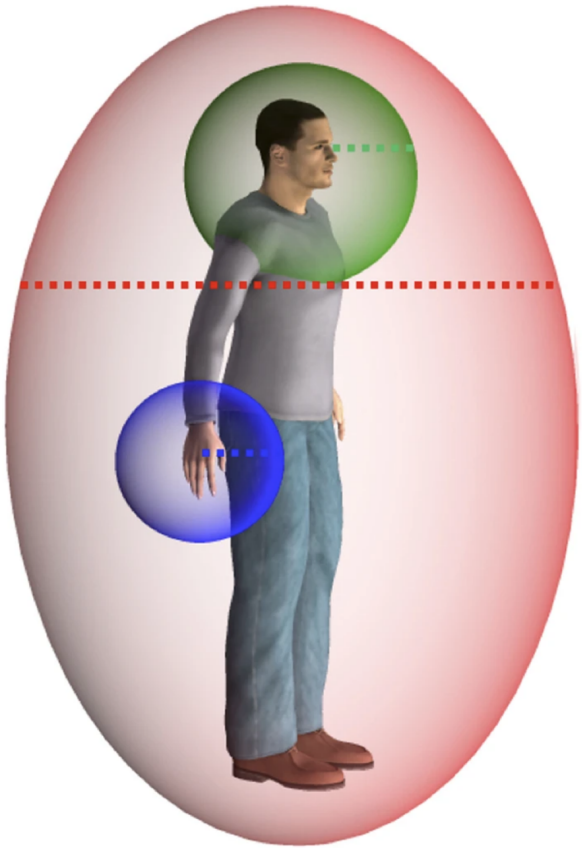

Figure 1.3: **Conceptual schema of the peri-hand (blue), peri-face (green) and peri-trunk (red) PPS representations.** Figure from [10]. [Under a Creative Commons Attribution 4.0 International License – `http://creativecommons.org/licenses/by/4.0/`]

make testable predictions, based on the properties of the model, for future empirical studies which can be compared with the model's behavior and properties. For example, our normative model of PPS encoding [17] showed that when a sensory uncertainty of approaching stimulus is increased (e.g., by changing light conditions), the PPS boundary becomes more graded. This can be experimentally tested and used for possible rejection of the model in the current form.

1.2 Related PPS encoding models

In this section, we will introduce related PPS encoding models and their connection to predictive PPS encoding.

To begin with, we will introduce the PPS model developed by Magosso et al. [18] and its various extensions [19–22]. The initial version of the computational model of PPS used a biologically motivated neural network with hardwired receptive fields of multisensory neurons to cover the entire tactile modality and the corresponding part of visual space, with some extension beyond the part covered by the tactile modality. The model was further extended by Hebbian learning to account for PPS extension after tool-use [19, 22]. Another variant of the model addressed short-term recalibration of the PPS [21]. A neural network model by Noel et al. [20] addressed the expansion of PPS with increasing speed of the stimulus approaching the body (see e.g., [23]). They extended the model of Magosso et al. [18] to

include neural adaptation, which led to a decrease in neural activation for slower stimuli and thus a decrease in the size of the PPS. The focus of these models was on potential neural mechanisms of PPS encoding, without being explicitly tailored for predictive PPS encoding. The models link visual stimuli close to the body with the tactile modality—a visual stimulus, indirectly via multimodal neurons and feedback connections, preactivates corresponding unimodal tactile neurons that can be more easily activated. Although, this can be seen as visuo-tactile prediction, the ability to predict future impact was not directly evaluated. Therefore, it is unclear whether the models can effectively use, for example, the previous trajectory of the stimulus to more accurately predict future impact.

Another neural network model proposed by Bertoni et al. [24] is based on the Restricted Boltzmann Machine (RBM). The model performs multisensory integration of tactile, proprioceptive and visual modalities. This model by Bertoni et al. is unique in that it takes into account proprioception—the PPS is anchored to the hand. In contrast to predictive PPS encoding, the model focuses mainly on spatial visuo-tactile prediction (related visual and tactile stimuli occur simultaneously during training) and ignores the temporal dimension. This limits the extension of PPS further away from the body.

Bufacchi et al. [25] created a geometric model of the PPS related to the hand blink reflex (see [26]) defensive response. They proposed that around a body area, the PPS is encoded as a probability field whose values correspond to the probability of the stimulus hitting the body area. The field is expected to be used to modulate the defensive response. The hand blink reflex was used to fit the model to measured data. Since the field reflects the probability of a future hit, the model also supports the idea that PPS encoding is related to visuo-tactile prediction. On the other hand, the model did not take into account dynamic properties of the stimulus (e.g., direction of motion). However, the consideration of dynamic properties may not be necessary for the prediction related to the hand blink reflex.

Another model was developed by Roncone et al. [27]. The PPS representation model was trained using the real humanoid robot iCub [28]. The probabilistic form (likelihood of contact) of the PPS representation was learned from looming objects that eventually hit the body. The model used stimulus distance and estimation of time to contact for each tactile unit to estimate the likelihood of contact. The distributed PPS representation was used for avoidance and reaching tasks with the real robot. This model performs visuo-tactile prediction to represent PPS in a different way than our models, but is conceptually close to them.

Recently, a model of PPS-like encoding based on reinforcement learning (RL) was proposed by Bufacchi et al. [29] (currently under review). The authors demonstrated that only two assumptions—(i) agents experience rewards (both positive and negative) when they get in contact with objects from the environment, (ii) they maximize reward by performing proper actions—are sufficient to explain most of the empirical properties of PPS representations. Moreover, they showed that impact prediction is an emergent property of the model representation. In other words, they suggested that the brain maximizes reward by choosing appropriate actions, and impact prediction is only a by-product of this process. The potential contribution of this new concept of PPS encoding is great. However, further research and critical exploration is needed to decide whether this more general approach is indeed a better modeling schema for PPS-related phenomena.

1.3 Roadmap of the thesis

This thesis is a compilation of three published journal articles and a conference article. Moreover, a preliminary results report of a work, which is a continuation of the published articles and currently in preparation for submission, is included. To introduce a story line of our

exploration of the mechanisms of PPS representations, this section unfolds how the presented works build on each other and how they are interrelated. Main contributions of each work are elaborated in Chapter 2. A roadmap of the thesis is in Fig. 1.4.

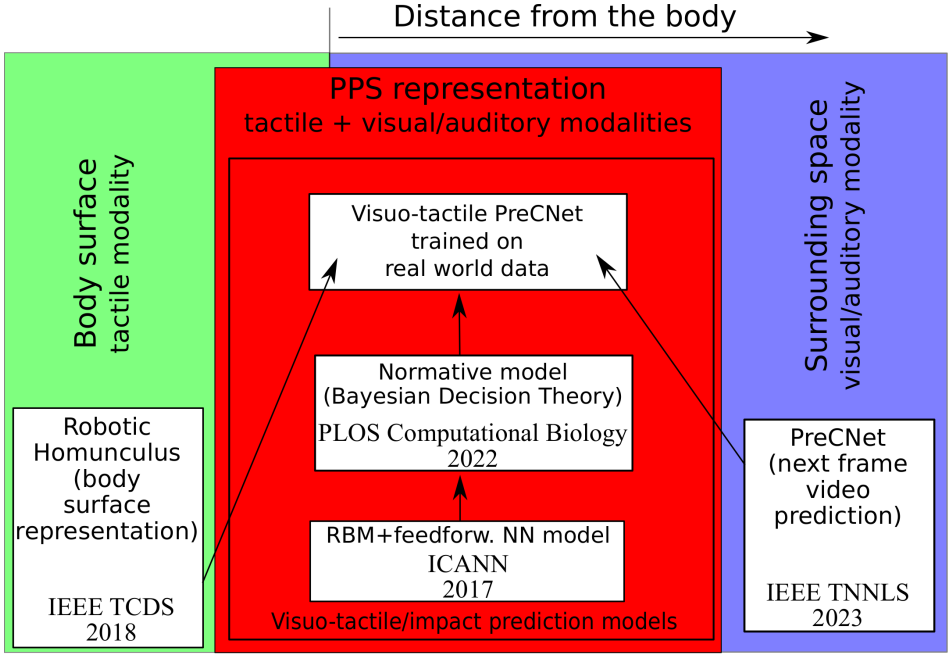

Figure 1.4: **Roadmap of the thesis.** The arrows show that the article at the end of the arrow builds on the article at the beginning of the arrow or that insights and weaknesses of one work influenced the following work (see the text for details). The green rectangle corresponds to the body surface, the blue one to the surrounding space. The red rectangle represents PPS which covers the body surface and a close subpart of the surrounding space. The white rectangles correspond to the different works.

We initially developed a biologically-inspired representation of the large areas of the body surface of the humanoid robot iCub [28] that is equipped with the pressure-sensitive tactile modality (see Fig. 1.5). This was motivated by the fact that the body surface is also a part of the PPS encoding—receptive fields of observed PPS related neurons commonly include a tactile area and its close surroundings [23], for example. Modifying a well known neural network architecture self-organizing map (SOM; [30]), we get 2D maps with representation of the large areas of the body surface which locally preserved topological organization of the tactile modality and performed compression of the tactile information [31] (see Fig. 1.5). The topological organization cannot be preserved globally, because the 3D surface of the robot (e.g. a cylindrical surface) cannot be reduced to 2D map with complete preservation of the topological organization. Although we have not directly used these maps in our PPS research yet, it had an impact on the choice of using 2D tactile topology preserving tactile maps with compression for tactile encoding in our following work—visuo-tactile PreCNet (see Chapter 3).

Our initial model of PPS representation departed from a Restricted Boltzmann Machine based neural network [32] for integration of the position and velocity and a feedforward neural network for the prediction itself (see Fig. 1.6). Because the model based on predictive PPS encoding successfully replicated the phenomenon of PPS expansion with increasing speed of an oncoming stimulus [23], the work was encouraging for the idea of predictive PPS encoding. However, the complexity of the model—it consists of two different neural network models—

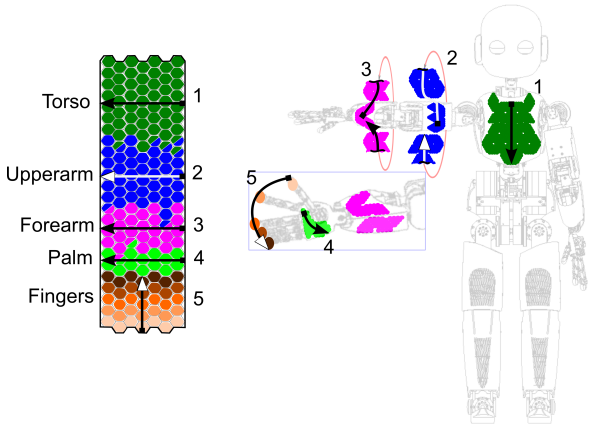

Figure 1.5: **Representation of the body surface of a humanoid iCub by a 2D tactile map.** (Left) A 2D tactile map whose neurons have receptive fields which cover the tactile modality. The map locally preserves tactile topology and compresses tactile information. (Right) A humanoid robot iCub with depicted tactile modality. Figure taken from [31]. ©2017 IEEE

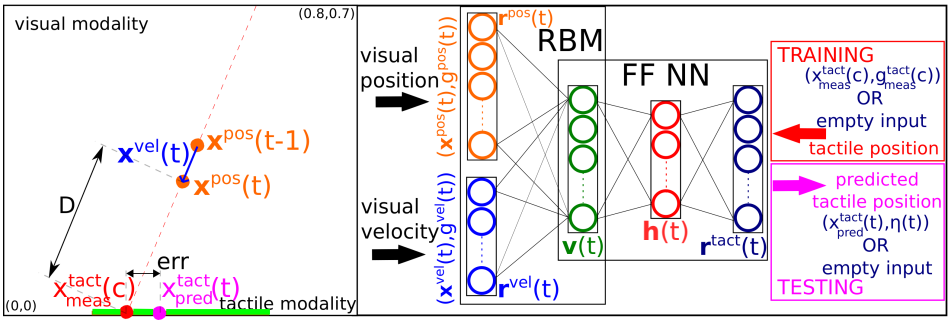

Figure 1.6: **Scenario and architecture.** (Left) 2D experimental scenario. Stimulus trajectory in orange; positions of stimulus at two different discrete time moments shown. "Skin" in green. (Right) Architecture of the neural network and illustration of training and testing (predicting) process. Figure and caption taken from [33].

and the fact that certain aspects of the model were based on more hypothetical mechanisms (e.g., existence of a sensory buffer) made it harder to relate and determine consequences of the results for the biological side of PPS research. This motivated us to propose a normative model of PPS, which is more transparent and based on more biologically plausible mechanisms.

We developed a normative model as the second predictive PPS encoding model. This model is based on the estimation of the probability of a moving object hitting the body and Bayesian Decision Theory. Although, the idea of the model is general, in this work, the focus of the model was on an experimental scenario with an object with a uniform linear motion. Since this setup is very common in cognitive neuroscience studies of PPS (e.g., [2,10]), it facilitated a comparison between the model's properties and the empirical results. This comparison revealed that the model replicated a wide range of empirically established characteristics of PPS.

However, this model was not intended for learning from data, which limits its use for investigating the development of PPS encoding with visuo-tactile data from child-caregiver-like interaction with a child-like robot. This led us to propose a deep neural network model for

next visuo-tactile frame prediction which also minimizes weighted prediction loss. In contrast to the previous model, the calculation is not hard-wired but learned from the training data. As a first step, we ignored the tactile modality and created a deep neural network for next frame video prediction based on predictive coding.

In next frame video prediction task a sequence of images is given and the task is to anticipate the succeeding image (see Fig. 1.7). We created a deep neural network PreCNet [36]

Figure 1.7: **Next frame video prediction task** A sequence of images is given as an input. The task is to predict the next image as close as possible to the ground truth image. The images taken from Caltech Pedestrian Dataset [34, 35] which was used for evaluation of the model.

based on predictive coding schema proposed by Rao and Ballard [37]. The state-of-the-art performance of the network on widely used datasets with images from a car-mounted camera was a good prerequisite for sufficient predictive performance of the model with real world child-caregiver-like data as inputs.

The network PreCNet was extended by a tactile modality and applied for predicting future visuo-tactile frames (see Fig. 1.8 for a visuo-tactile frame). As a part of loss function related

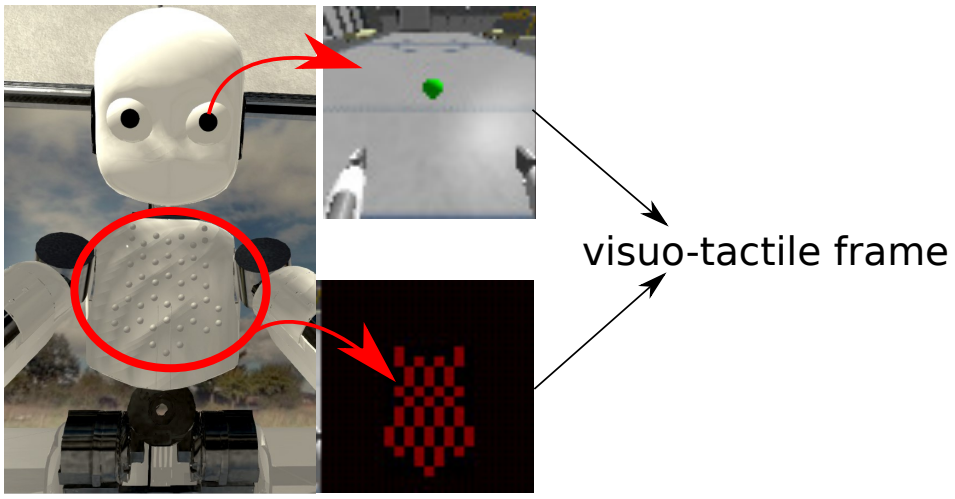

Figure 1.8: **Visuo-tactile frame.** An image from the left camera of iCub and a tactile map representing the torso are integrated into a visuo-tactile frame. The images were created in Neurorobotic Platform simulator [38], the skin was created using [39] as a starting point.

to tactile channel, a generalized version of the loss function of the normative model [17] was

used. The model was trained using data which mimicked a real interaction between a baby and a caregiver by using a child-like humanoid robot iCub (see Fig. 1.9). The initial results

Figure 1.9: **Real world interaction with a child-like robot iCub.** The person walks up to the robot and strokes it, much like a parent would do it.

showed that the visuo-tactile PreCNet can predict future tactile activations. Although this work is still ongoing, we decided to incorporate it into the thesis in the form of a brief ongoing work report (see Chapter 3). This aims to clarify the connection between our previous work, especially the network for next frame video prediction, and the main topic of the thesis.

The thesis is structured as follows. Chapter 2 presents main contributions of the individual published works. Chapter 3 is dedicated to the ongoing work report of the visuo-tactile PreCNet. The next chapter contains the conclusions (Chapter 4). Chapter 5 is devoted to discussion and future work.

Chapter 2

Contribution of published works

This chapter outlines the main contributions of each article that constitutes the thesis. We provide an overview of each model used and summarize the main results. The contributions of the individual articles are compiled in Table 2.1.

2.1 Robotic homunculus: Learning of artificial skin representation in a humanoid robot motivated by primary somatosensory cortex

Hoffmann, M., Straka, Z., Farkaš, I., Vavrečka, M. and Metta, G., 2017. 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), pp.163-176.

The article is in Appendix A. The code is available at `https://github.com/matejhof/robotic-homunculus-supporting-materials`.

We used a humanoid robot, iCub [28], equipped with an artificial pressure-sensitive skin to investigate how the representation of large areas of the body surface, which mimics that found in the primary somatosensory cortex of primates [40, 41], can be obtained from local tactile stimulations of the humanoid's body (see Fig. 2.1).

We employed a well-known self-organizing map (SOM; [30]), which is known to preserve the topology of the input data, and investigated the arrangement of the trained maps. We found that these fully data-driven maps have very different and variable arrangements in contrast to the arrangement of primary somatosensory cortex (see Fig. 2.2 left part). In addition, the standard SOM was unable to handle multiple concurrent tactile stimulations. Motivated by findings from biology that the arrangement of the cortical sheet seems to be roughly genetically determined (see [31] for details), we modified the standard SOM by adding the possibility to restrict the receptive fields of individual neurons within the map and created the Self-Organizing Map With Maximum Receptive Field Size Setting (MRF-SOM). This allowed us to create a map—learning was still used—with an appropriate arrangement, as shown in Fig. 2.2 (right), and also significantly increased the robustness of the network to multiple simultaneous tactile stimulations.

From an application point of view, the method can be used for bio-inspired representations of large-area tactile arrays [43] or the representation of proprioceptive inputs [44], for example.

Table 2.1: Contributions summary of each work in three key areas (see Chapter 1).

	Understanding of PPS/body surface representation	**Computational methods (only contributions here)**	**Applications**
Robotic homunculus [31]; Appendix A	Effect of activity-dependent vs. independent factors on somatotopic map formation.	Modification of SOM that allows to set Maximum Receptive Field Size of individual neurons.	Bio-inspired representations of large-area tactile arrays or the representation of proprioceptive inputs.
RBM and FFNN model of PPS [33]; Appendix B	Visuo-tactile prediction as a mechanism of PPS representation learning is feasible.	No significant contribution.	No significant contribution.
Normative model of PPS representation [17]; Appendix C	The normative model of PPS based on impact prediction mechanism replicates many of the PPS characteristics. This supports the idea of the "predictive PPS encoding". Predictions for future empirical work are proposed.	No significant contribution.	No significant contribution.
PreCNet [36]; Appendix D	No significant contribution.	Deep neural network based on a predictive coding schema.	State-of-the-art performance for next frame video prediction task.

2.2 Learning a peripersonal space representation as a visuo-tactile prediction task

Straka, Z. and Hoffmann, M., 2017. Learning a peripersonal space representation as a visuo-tactile prediction task. In Artificial Neural Networks and Machine Learning–ICANN 2017: 26th International Conference on Artificial Neural Networks, Alghero, Italy, September 11-14, 2017, Proceedings, Part I 26 (pp. 101-109). Springer International Publishing.

The article is in Appendix B. The code is available at `https://github.com/ZdenekStraka/icann2017-pps`.

The model of PPS, in a 2D environment (see Fig. 1.6), is based on two parts: (i) an

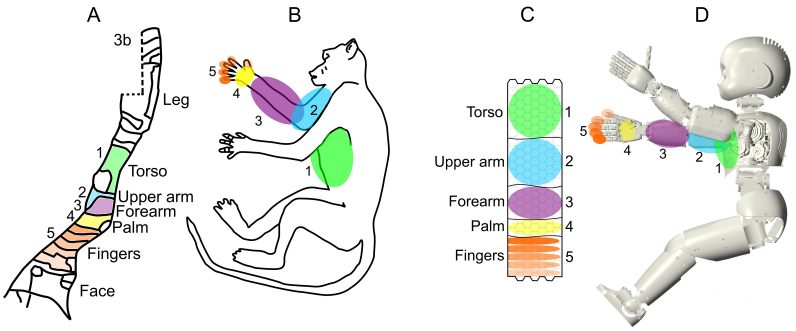

Figure 2.1: **Representation of tactile body surface in monkey and robot.** (A) and (B) Simplified representation of selected body parts in Brodmann area 3b of macaque monkey. Numbers and color code mark the correspondences between the cortical areas and skin surface on the body parts that will be modeled using the iCub robot. Redrawn and adapted after [42]. (C) and (D) Schematics of analogous situation in the robot—approximate target for the SOM algorithm. Figure and caption taken from [31]. ©2017 IEEE

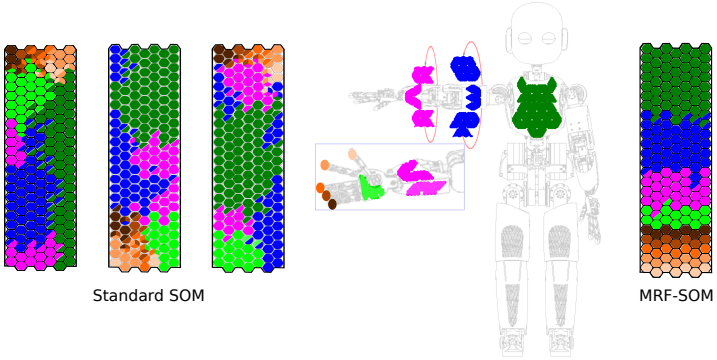

Figure 2.2: **Learned maps from tactile stimulation on right side of robot body.** (LEFT) Learned representations using **standard SOM**. The three panels depict the maps (24×7 neurons) after learning as a result of three runs of the algorithm on the same training set. The arrangement is variable and significantly different from that found in primary somatosensory cortex. (RIGHT) Learned representation using **MRF-SOM**. The arrangement resembles that found in primary somatosensory cortex (see Fig. 2.1). Figures and a caption from [31] used as a base for this figure. ©2017 IEEE

RBM-based neural network which integrates position and velocity and (ii) a feedforward neural network for visuo-tactile prediction (see Fig. 1.6 for the architecture). In addition, all variables are encoded by the neurons with a "probabilistic population code" [32, 45] to incorporate uncertainty. The work showed that the idea of learning PPS representation as a tactile prediction from looming visual stimuli is feasible. Additionally, the model showed PPS expansion with an increasing speed of the looming stimuli, which is an empirically observed property of PPS representations [23].

2.3 A normative model of peripersonal space encoding as performing impact prediction

Straka, Z., Noel, J.P. and Hoffmann, M., 2022. A normative model of peripersonal space encoding as performing impact prediction. PLOS Computational Biology, 18(9), p.e1010464.

The article is in Appendix C. The code is available at `https://github.com/ctu-vras/pps-normative-model`.

The normative model that links PPS encoding with impact prediction is based on estimation of the probability of a moving object hitting the body (see "Hit probability estimation" in Fig. 2.3) and Bayesian Decision Theory (see "Bayesian decision" in Fig. 2.3). The model allows for the consideration that the cost of not predicting an impact when it occurs may be greater than the cost of predicting an impact when it does not occur.

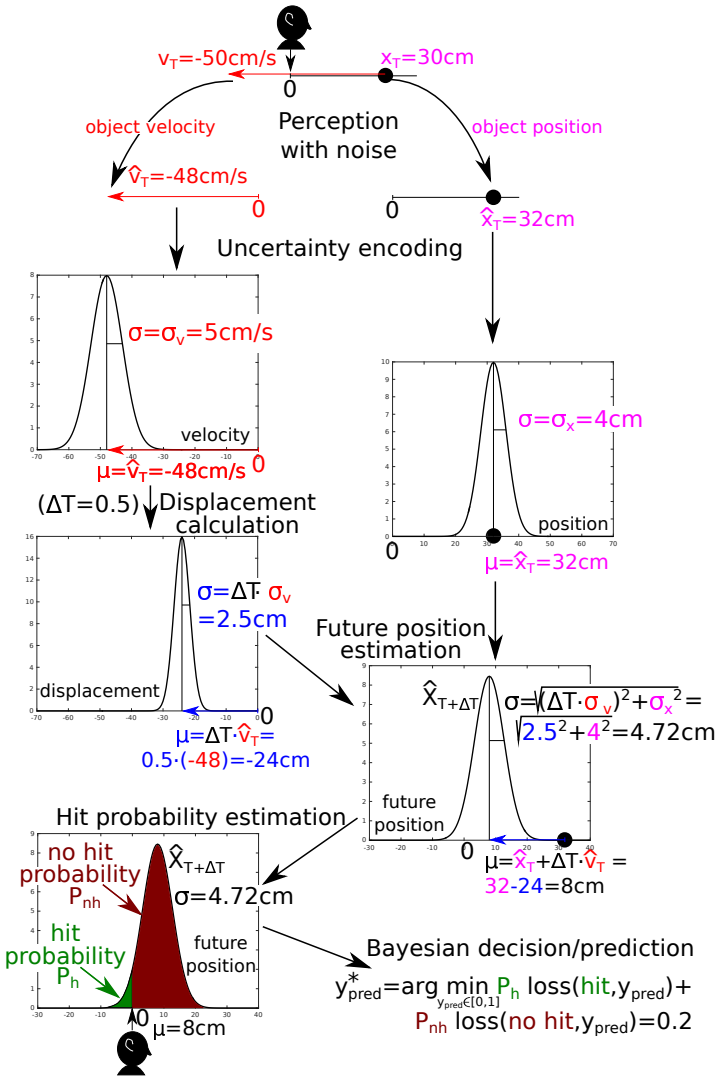

Figure 2.3: **Schema of the normative model of PPS with an example of calculation.** Figure from [17].

The hypothesis that PPS reflects impact prediction is supported by the results of the

model, which replicated the following empirically determined characteristics of PPS: (i) defines a gradual boundary between near and far space, (ii) shows expansion of PPS as the speed of an incoming object increases, (iii) demonstrates larger PPS for a looming than a receding stimulus, (iv) scales according to numerically expressed characteristic of the environmental objects and (v) it can accommodate different PPS sizes for various parts of the body. In addition, we proposed specific predictions for future empirical work.

2.4 PreCNet: Next-frame video prediction based on predictive coding

Straka, Z., Svoboda, T. and Hoffmann, M., 2023. PreCNet: Next-frame video prediction based on predictive coding. IEEE Transactions on Neural Networks and Learning Systems. [in press]

The article is in Appendix D. The code is available at `https://github.com/ctu-vras/precnet`.

We transformed the seminal predictive coding schema proposed by Rao and Ballard [37] into a deep neural network (PreCNet) for next frame video prediction, while remaining maximally faithful to the original schema.

The network attained state-of-the-art performance on a widely used benchmark (KITTI [46] for training, Caltech Pedestrian Dataset [34,35] for testing) for next frame video prediction task, consisting of videos from a car-mounted camera (see Fig. 2.4 for a qualitative comparison of PreCNet with other state-of-the-art methods).

Furthermore, we showed that the use of a larger dataset (2M images subset of BDD100K [47]) than the standard one (KITTI with 41k images) yielded another significant improvement. This revealed the limitations of using KITTI dataset for training.

Figure 2.4: **Qualitative comparison of PreCNet with other state-of-the-art methods on the Caltech Pedestrian Dataset.** Ten input frames were given (see frames for t = 8, t = 10), and the next one (t = 11) was predicted (RC-GAN used only four input frames) by the models (for references, see [36]). Figure and caption (with modifications) from [36]. ©2023 IEEE

Chapter 3

Ongoing work: Visuo-tactile prediction during child-caregiver-like interaction

3.1 Visuo-tactile prediction during child-caregiver-like interaction

This work is currently being prepared for submission. Thus, this chapter takes the form of a brief ongoing work report. The focus of this report centers primarily on the motivation behind this work, its relation to our previous work (which forms the core of the thesis), and our plans for the remaining work. This work is a continuation of Adrian Pitonak's bachelor thesis [48], which I supervised.

3.1.1 Introduction

It is still unclear how PPS encoding is formed in the brain during early childhood. There is initial evidence that a simple form of (auditory) PPS encoding is present even a few hours after birth [49]. Additionally, the results of the study showed a significant positive correlation between the multisensory integration (calculated from EEG data) of the tactile and near auditory stimulation and postnatal age. This is supportive for the concept that PPS representations are not fixed and are still developing in early childhood, although more research needs to be done.

The complexity of babies' interaction with the environment—the movement of caregivers is stochastic, highly variable and non-uniform, for example—, which has likely impact on development, poses a limitation on the use of "disembodied" computational models for exploring the developmental process. Computational models mostly fail to reflect the interaction complexity. This work explores the possibility of using embodied robotic models and collecting sensory data during scenarios that resemble the natural interaction of a baby with a caregiver. A humanoid child-like robot, iCub [28], was used to collect data to train PPS representations.

This work has also an application aspect. Roncone et al. [27] created a white box—processing of the raw visual inputs consisted of several handcrafted stages, for example—model of PPS encoding. It was trained, using probabilistic estimation approach, to predict tactile contact from raw visual stimuli. The model was successfully used for avoidance and reaching tasks with the real robot. As deep learning approach often outperforms handcrafted methods, we employ a deep neural network, based on our previous work [36], for visuo-tactile prediction. This may improve prediction performance and therefore contribute to performance improvement of the robot in tasks such as avoidance and reaching.

3.1.2 Visuo-tactile predictions

The task is to predict a future tactile activation given an observed image sequence. This is demonstrated using an illustration with inputs from a simulated environment, as shown in Fig. 3.1. Our visuo-tactile prediction model extends PreCNet [36] by incorporating two-

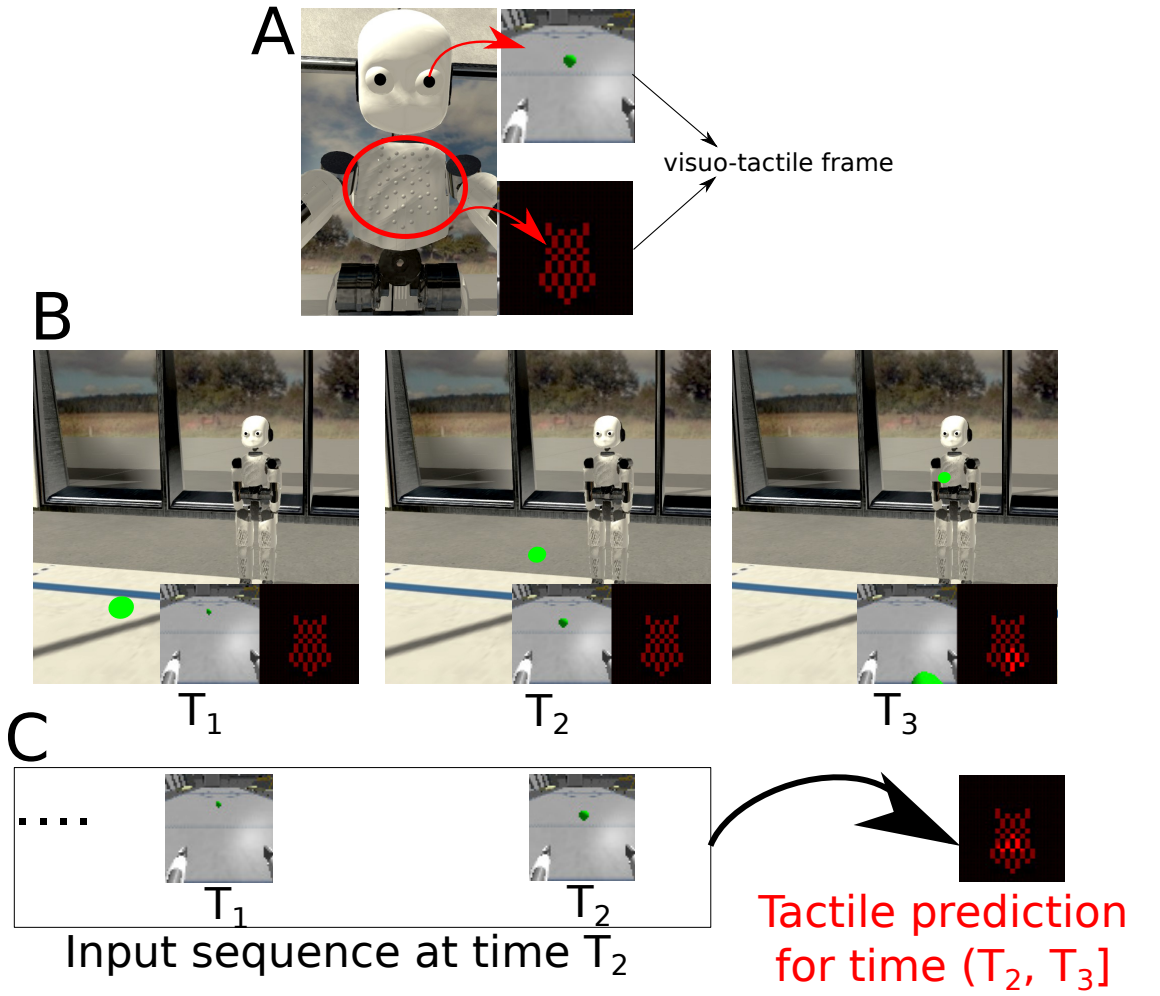

Figure 3.1: **Illustration of a visuo-tactile prediction task in a simulated environment. (A)** An image from the left camera of iCub and a tactile map representing the torso are integrated into a visuo-tactile frame. **(B)** A green ball is looming toward the robot iCub. The corresponding camera images and tactile maps are displayed side by side. The onset of tactile activation was between T_2 and T_3. **(C)** The task at time T_2 is to predict the tactile activation that will occur between time T_2 and T_3 from the input sequence of images. The images were created in Neurorobotic Platform simulator [38]; the skin was created using [39] as a starting point.

dimensional tactile maps that reflect the topology of the tactile modality into the input frames (see Fig. 3.1 A). In the case of the real robot, the tactile maps also perform compression of the tactile information. Furthermore, a part of a loss function related to the tactile modality was created as a generalization of the loss function used in our normative PPS model [17].

3.1.3 Preliminary Results

We trained the network on a dataset containing multiple sequences of a person walking towards and touching iCub (sequences without touching were also included). The motivation was to mimic the interaction between a child and a caregiver (see Fig. 3.2). The network was tested

Figure 3.2: **Real world interaction with a child-like robot iCub.** The person walks up to the robot and strokes it, much like a parent would do it.

on the test subset of the dataset. The network predicted future tactile stimuli given the input sequences, as shown in Fig. 3.3 and in a video [`https://drive.google.com/file/d/1PFT1jC_4TFcPduzASRnr0k2QZ1KhFLu3/`] (in slow motion [`https://drive.google.com/file/d/1kMm3dqPwu04p9FOqVfEsyeAxwmw-91mw/`]). The upper part of the video corresponds to the input frame, the lower part to the prediction of the future video-tactile frame. The model is expected to predict tactile activation before it happens, as demonstrated in Fig. 3.3 or in the movie.

The distance of the experimenter's right hand from the touch location on the iCub's torso, used (only) for evaluation, was obtained using a motion capture system. This allowed to obtain the dependence of the tactile prediction values on the distance of the hand from the robot (see Fig. 3.4). Assuming that the tactile prediction value is closely related to PPS encoding, the PPS response from Fig. 3.4 can be used to compare the model with empirical properties of PPS, in a similar fashion to [17].

In addition, we also used a synthetic dataset (see Fig. 3.1) created in the Neurorobotic Platform (NRP) simulator [38] and trained and tested the network using the dataset. Certain properties of the model can be analyzed more easily in a controlled environment where an object moves uniformly and linearly towards the robot.

3.1.4 Future Work

There are several questions we would like to address in this work. First, we want to explore the idea of PPS encoding as a visuo-tactile prediction in a developmental scenario. Specifically, it is being questioned whether the idea that interactions in which a caregiver approaches a baby are the basis for the development of PPS encoding is plausible and can lead to PPS representations with empirically observed properties. We also aim to propose testable predictions for future empirical studies.

A synthetic scenario involving a looming ball heading towards a robot was created in NRP to train the model and gain insight into some properties of the model, such as the effect of the speed of the moving object on the tactile predictions. The properties of the model will be analyzed.

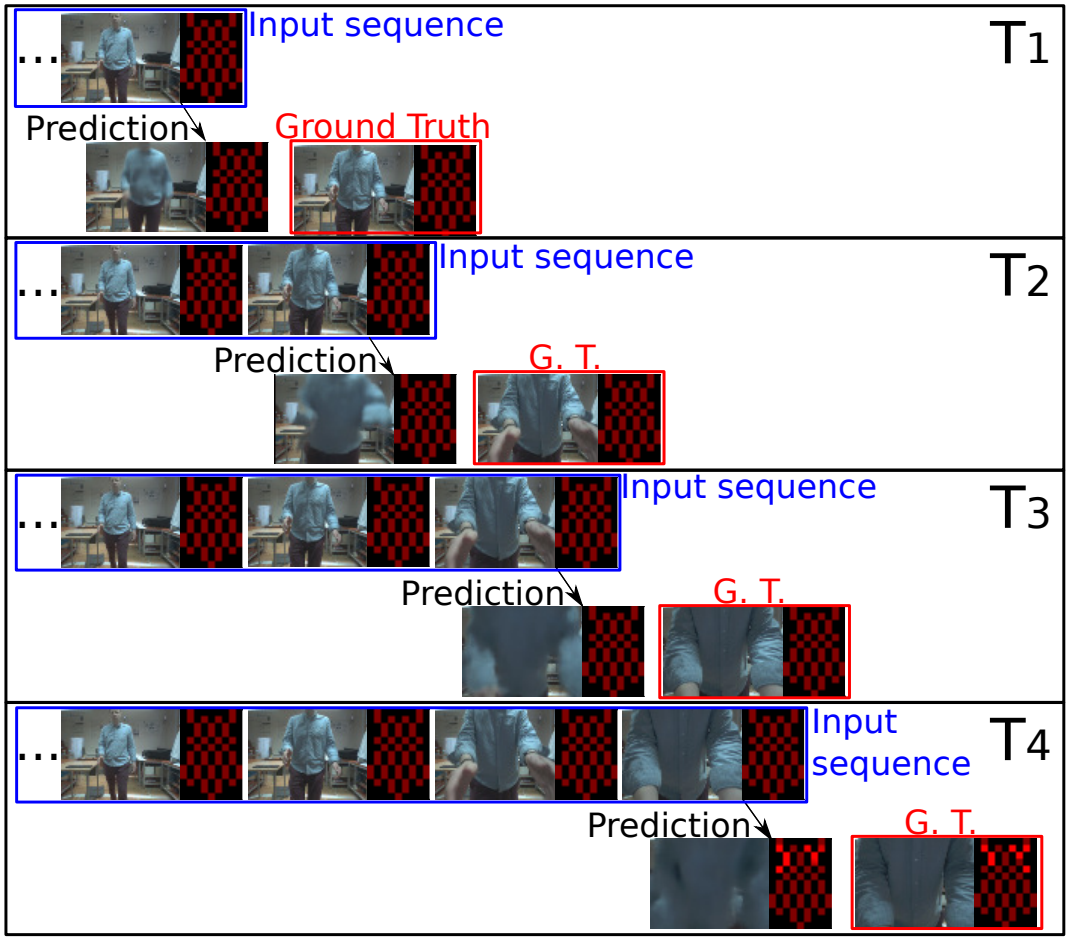

Figure 3.3: **Visuo-tactile predictions of the network.** Four consecutive input visuo-tactile sequences and corresponding predictions are shown. After inputting a sequence, a prediction of the visuo-tactile frame is generated. The predicted frame is expected to be close to the following input frame (ground truth). Although this work focuses on tactile prediction, the prediction of the next visual frame is also obtained as a by-product.

The model may also have an important contribution in the field of robotics. We will study whether this data-driven model can be used for contact prediction in the context of human-robot interaction or as a reliable method for collision avoidance. For example, the concept of self-supervised learning of safety zones around a robot during its interaction with the environment will be studied.

Another potential contribution of this work is the unique method of extending visual frames with tactile maps to form visuo-tactile frames. Preliminary results indicate that the method is effective and could have applications beyond our model. This will be investigated further.

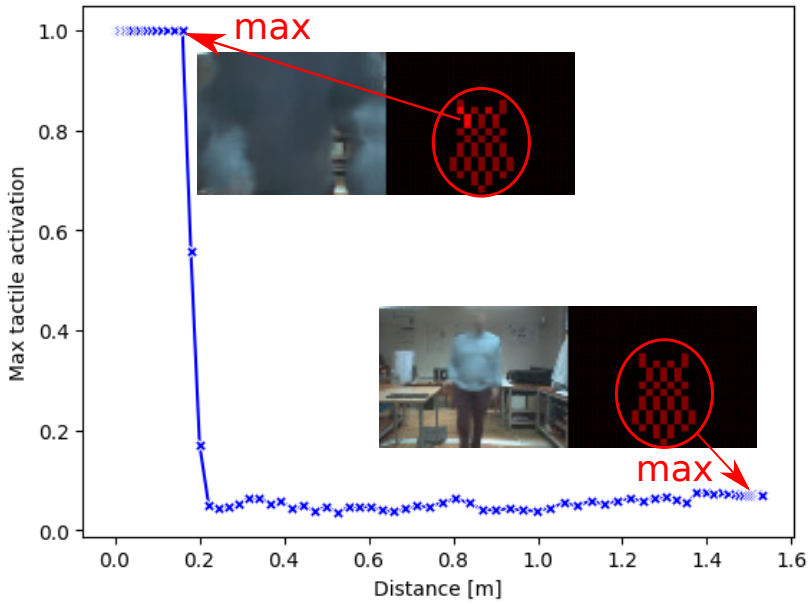

Figure 3.4: **Dependence of a predicted tactile activation on the distance of the experimentator's hands from iCub.** The network predicted negligible tactile activation (see the lower tactile map) when the looming hands were more than about 20 cm away from the robot. When the distance was less than 20 cm, tactile activation was predicted as shown in the upper tactile map. The maximum tactile activation over all pixels of the predicted tactile map was used in the plot.

Chapter 4

Conclusions

The focus of the thesis was primarily on the modeling of peripersonal space (PPS) encoding and biologically motivated representations of the body surface. The exploration of the encoding of PPS was performed using the concept of visuo-tactile prediction. The code of all our models has been made publicly available.

We began by modifying a self-organizing map neural network to incorporate prior knowledge about the representation of the body surface in the primary somatosensory cortex. We used tactile data from the humanoid robot iCub to investigate the process of formation of this representation [31].

In the following work, we proposed a neural network-based architecture that focused mainly on the feasibility of the idea of visuo-tactile prediction as the key mechanism of PPS encoding [33]. This network replicated the empirical observation of PPS expansion that occurs with increasing velocity of a looming stimulus. In subsequent work, we proposed a normative model of PPS encoding to perform impact prediction [17]. Using Bayesian decision theory, the model was able to replicate many characteristics of PPS. Suggestions for future empirical studies based on the properties of the model are an important part of the paper.

To explore the development of PPS encoding using raw visuo-tactile inputs during an interaction between a humanoid robot and a caregiver that mimics the interaction between a child and a caregiver, we first developed a deep neural network (PreCNet) for next frame video prediction based on the predictive coding schema [36]. PreCNet achieved state-of-the-art performance on a commonly used benchmark for next-frame video prediction. Currently, we are preparing a manuscript where the visual frames of PreCNet are extended by adding a tactile modality. The network is then employed for visuo-tactile prediction. We will analyze the PPS encoding properties of the model, particularly in the context of PPS development. We will also explore the application of this method in robotics. This research is presented in the thesis as an ongoing work report.

Chapter 5

Discussion

In this chapter, the discussion will be primarily related to the concept of peripersonal space (PPS) encoding as visuo-tactile prediction (*predictive PPS encoding*), which is the central focus of the thesis. Since detailed discussions can be found in each published paper, we will avoid discussing them in depth here and instead focus on general aspects that were addressed in the thesis.

The concept of predictive PPS encoding appears to be a natural candidate for explaining the properties of PPS representations (e.g., [13, 14]) and is widespread in the scientific community (see [15, 16] for reviews). Therefore, it was anticipated that the models' properties would generally align with empirical findings. This largely proved to be the case. Given the generality of the concept of predictive PPS encoding, there is a danger that an implementation of the concept will be overfitted to the desired properties. It may happen by a "proper" choice of model or training/test dataset parameters or their form. For instance, certain features of PPS could be explained by proposing a hypothetical frequent situation that a child might experience during his or her development, without clear evidence that the situation actually occurs significantly often. As there are limited options to avoid this situation without knowing the correct model or data and its parameters, it is necessary to be careful in interpreting the results. Our approach to address the "overfitting" issue predominantly involved three strategies. The first strategy is that we published concrete non-trivial properties of the model that can be observed in future empirical work (see [17]). This is an efficient test of the model since it is currently unknown whether these properties will indeed be observed. The second strategy is to test the model in several experimental scenarios (see [17]). If the model can replicate PPS responses from more scenarios, the risk that it is caused by overfitting is smaller. The third strategy is to utilize bio-plausible datasets (see Chapter 3). This was the motivation for the third model, which was trained on a visuo-tactile dataset that mimics caregiver-child interactions using a humanoid robot. Although the dataset does not correspond to the child's actual sensory experience, it is based on real data, which may help to increase its similarity to the child's actual sensory experience in some aspects.

Our focus was mostly on the high-level functional mechanism rather than the mechanistic explanation of how neural circuits implement the predictive encoding of PPS. It is a challenging task to find a biologically plausible neural network model (preferably based on spiking neural networks) of the predictive PPS encoding that covers all important aspects, including velocity and sensory uncertainty. As explained in Section 1.2, Magosso et al.'s biologically motivated neural network PPS encoding model [18] can be seen as performing a visuo-tactile prediction by linking visual stimuli close to the body with the tactile modality. Moreover, Noel et al. [20] expanded the model by adding neural adaptation to increase the size of the PPS with increasing speed of the approaching stimulus, and suggested that the neural adaptation

mechanism may be an important part of a related predictive mechanism in the brain that is presumably involved in the addressed phenomenon. However, the predictive abilities of the models have not been explicitly investigated. Consequently, it is unclear whether the models can effectively use, for example, the previous stimulus trajectory for accurate future contact prediction. Investigating the predictive abilities of these models, and potentially extending them to perform more accurate future tactile predictions, may help bridge the gap between a biologically plausible neural architecture and predictive PPS encoding mechanisms.

If visuo-tactile predictions are indeed a key mechanism in PPS encoding, their primary purpose is likely related to the survival of the organism. Consequently, the predictions must be evident at the behavioral level. As different actions have varying time courses and durations (e.g., blinking vs. sidestepping), the brain needs to make predictions with different time steps based on the related actions. This was reflected in our normative model [17] and visuo-tactile PreCNet (see Chapter 3) by choice of predictive time step. When we compared the predicted tactile impact values produced by the models with empirically observed PPS responses like modulation of reaction time—presumably related to defensive reactions—we assumed that a higher value of predicted contact would result in a stronger modulation of the potential model's PPS response. Given the seemingly straightforward connection between predicting a threatening stimulus and eliciting a defensive reaction, it is reasonable to assume that the close link between the prediction and reaction exists. Therefore, we performed a direct comparison of the properties of the predicted impact values with the properties of the measured responses. However, the precise relationship between prediction and response is still unclear and likely depends on the individual defensive reaction. Consequently, a possible next step in investigating predictive PPS encoding would be to incorporate actions into the models, similar to Roncone et al.'s extension of their predictive PPS encoding robotic model with a controller that utilized predictions during reaching and avoidance tasks [27]. For cognitive neuroscience research on PPS encoding, it would be particularly intriguing to expand the predictive PPS encoding model to encompass a basic defensive reaction, such as protective blinking. By doing so, it would be possible to compare behavior-based responses of the model, like reaction times, with observed responses in an empirical experiment. The utilization of a simpler controller should result in more transparent inquiry and simplified analysis of the findings. Alternatively, Bufacchi et al. [25] conducted a direct mapping of predictions onto PPS response modulations. While this approach allows for a straightforward comparison between the predictive PPS encoding model and the organism's PPS encoding, using a model designed for actual defensive actions is a more technically demanding yet bio-plausible approach.

Chapter 6

Publications

Main publications from my studies are organized into categories and listed in reverse chronological order within each category. Impact factor (IF) values are taken from *Journal Citation Reports* for the year 2022 (Clarivate, 2023). Each publication is accompanied by author share.

Thesis subject-related publications

Impacted journal publications

1. Straka, Z.; Svoboda, T.; Hoffmann, M.: PreCNet: Next-frame video prediction based on predictive coding.In: *IEEE Transactions on Neural Networks and Learning Systems*, IEEE, 2023. [to appear] [IF=10.4]
   citations: 4 in Web of Science (WoS), 11 in Google Scholar (GS)

   Straka, Z. (50 %); Svoboda, T. (25 %); Hoffmann, M. (25 %)

2. Straka, Z.; Noel, J. P.; Hoffmann, M.: A normative model of peripersonal space encoding as performing impact prediction. In: *PLOS Computational Biology*, Public Library of Science, 2022, 18(9). [IF=4.3]
   citations: 2 in GS

   Straka, Z. (50 %); Noel, J. P. (25 %); Hoffmann, M. (25 %)

3. Hoffmann, M.; Straka, Z.; Farkaš, I.; Vavrečka, M.; Metta, G.: Robotic homunculus: Learning of artificial skin representation in a humanoid robot motivated by primary somatosensory cortex. In: *IEEE Transactions on Cognitive and Developmental Systems*, IEEE, 2017, 10(2), pp. 163-176. [IF=5.0]
   citations: 19 in WoS, 38 in GS

   Hoffmann, M. (32 %); **Straka, Z. (32 %, equal contribution with M.H.)**; Farkaš, I. (16 %); Vavrečka, M. (10 %); Metta, G. (10 %)

Conference proceedings (excerpted by WoS)

1. Straka, Z.; Hoffmann, M.: Learning a peripersonal space representation as a visuotactile prediction task. In: *26th International Conference on Artificial Neural Networks (ICANN)*, Springer International Publishing, 2017, Proceedings, Part I pp. 101-109. [**ENNS Best Paper Award**]
   citations: 6 in WoS, 9 in GS

   Straka, Z. (50 %); Hoffmann, M. (50 %)

Publications not related to thesis subject

1. Svarny, P.; Straka, Z.; Hoffmann, M.: Versatile Distance Measurement between Robot and Human Key Points Using RGB-D Sensors for Safe HRI In: *1st Workshop on Proximity Perception in Robotics at IROS 2018.* KIT Scientific Publishing, 2018.
   citations: 2 in GS

   Svarny, P. (70 %); **Straka, Z. (10 %)**; Hoffmann, M. (20 %)

2. Svarny, P.; Straka, Z.; Hoffmann, M.: Toward Safe Separation Distance Monitoring from RGB-D Sensors in Human-Robot Interaction In: *Proceedings of the International PhD Conference on Safe and Social Robots.* Strasbourg: Commission of the European Communities, 2018, pp. 11-14.
   citations: 13 in GS

   Svarny, P. (70 %); **Straka, Z. (10 %)**; Hoffmann, M. (20 %)

Bibliography

[1] N. Kriegeskorte and P. K. Douglas, "Cognitive computational neuroscience," *Nature neuroscience*, vol. 21, no. 9, pp. 1148–1160, 2018.

[2] M. S. Graziano, X. T. Hu, and C. G. Gross, "Visuospatial properties of ventral premotor cortex," *Journal of neurophysiology*, 1997.

[3] S.-M. Khaligh-Razavi and N. Kriegeskorte, "Deep supervised, but not unsupervised, models may explain it cortical representation," *PLoS computational biology*, vol. 10, no. 11, p. e1003915, 2014.

[4] D. L. Yamins, H. Hong, C. F. Cadieu, E. A. Solomon, D. Seibert, and J. J. DiCarlo, "Performance-optimized hierarchical models predict neural responses in higher visual cortex," *Proceedings of the national academy of sciences*, vol. 111, no. 23, pp. 8619–8624, 2014.

[5] C. F. Cadieu, H. Hong, D. L. Yamins, N. Pinto, D. Ardila, E. A. Solomon, N. J. Majaj, and J. J. DiCarlo, "Deep neural networks rival the representation of primate it cortex for core visual object recognition," *PLoS computational biology*, vol. 10, no. 12, p. e1003963, 2014.

[6] R. Pfeifer and C. Scheier, *Understanding intelligence*. MIT press, 2001.

[7] G. Rizzolatti, C. Scandolara, M. Matelli, and M. Gentilucci, "Afferent properties of periarcuate neurons in macaque monkeys. ii. visual responses," *Behavioural brain research*, vol. 2, no. 2, pp. 147–163, 1981.

[8] G. Rizzolatti, L. Fadiga, L. Fogassi, and V. Gallese, "The space around us," *Science*, vol. 277, no. 5323, pp. 190–191, 1997.

[9] M. S. Graziano and D. F. Cooke, "Parieto-frontal interactions, personal space, and defensive behavior," *Neuropsychologia*, vol. 44, no. 13, pp. 2621–2635, 2006.

[10] A. Serino, J.-P. Noel, G. Galli, E. Canzoneri, P. Marmaroli, H. Lissek, and O. Blanke, "Body part-centered and full body-centered peripersonal space representations," *Scientific reports*, vol. 5, no. 1, p. 18603, 2015.

[11] J. Cléry and S. B. Hamed, "Functional networks for peripersonal space coding and prediction of impact to the body." in *The world at our fingertips*, F. de Vignemont, A. Serino, H. Y. Wong, and A. Farnè, Eds. Oxford University Press, 2021, pp. 61–79.

[12] A. Serino, "Peripersonal space (PPS) as a multisensory interface between the individual and the environment, defining the space of the self," *Neuroscience & Biobehavioral Reviews*, vol. 99, pp. 138–159, 2019.

[13] J. Cléry, O. Guipponi, S. Odouard, C. Wardak, and S. B. Hamed, "Impact prediction by looming visual stimuli enhances tactile detection," *Journal of Neuroscience*, vol. 35, no. 10, pp. 4179–4189, 2015.

[14] M. K. Huijsmans, A. M. de Haan, B. C. Müller, H. C. Dijkerman, and H. T. van Schie, "Knowledge of collision modulates defensive multisensory responses to looming insects in arachnophobes." *Journal of Experimental Psychology: Human Perception and Performance*, vol. 48, no. 1, p. 1, 2022.

[15] J. Cléry and S. B. Hamed, "Frontier of self and impact prediction," *Frontiers in psychology*, vol. 9, p. 1073, 2018.

[16] H. Dijkerman and W. Medendorp, "Visuotactile predictive mechanisms of peripersonal space," in *The world at our fingertips: a multidisciplinary exploration of peripersonal space*. Oxford University Press, 2021, pp. 81–100.

[17] Z. Straka, J.-P. Noel, and M. Hoffmann, "A normative model of peripersonal space encoding as performing impact prediction," *PLOS Computational Biology*, vol. 18, no. 9, p. e1010464, 2022.

[18] E. Magosso, M. Zavaglia, A. Serino, G. Di Pellegrino, and M. Ursino, "Visuotactile representation of peripersonal space: a neural network study," *Neural computation*, vol. 22, no. 1, pp. 190–243, 2010.

[19] E. Magosso, M. Ursino, G. di Pellegrino, E. Làdavas, and A. Serino, "Neural bases of peri-hand space plasticity through tool-use: Insights from a combined computational–experimental approach," *Neuropsychologia*, vol. 48, no. 3, pp. 812–830, 2010.

[20] J.-P. Noel, O. Blanke, E. Magosso, and A. Serino, "Neural adaptation accounts for the dynamic resizing of peripersonal space: evidence from a psychophysical-computational approach," *Journal of neurophysiology*, vol. 119, no. 6, pp. 2307–2333, 2018.

[21] J.-P. Noel, T. Bertoni, E. Terrebonne, E. Pellencin, B. Herbelin, C. Cascio, O. Blanke, E. Magosso, M. T. Wallace, and A. Serino, "Rapid recalibration of peri-personal space: psychophysical, electrophysiological, and neural network modeling evidence," *Cerebral Cortex*, vol. 30, no. 9, pp. 5088–5106, 2020.

[22] A. Serino, E. Canzoneri, M. Marzolla, G. Di Pellegrino, and E. Magosso, "Extending peripersonal space representation without tool-use: evidence from a combined behavioral-computational approach," *Frontiers in behavioral neuroscience*, vol. 9, p. 4, 2015.

[23] L. Fogassi, V. Gallese, L. Fadiga, G. Luppino, M. Matelli, and G. Rizzolatti, "Coding of peripersonal space in inferior premotor cortex (area F4)," *Journal of neurophysiology*, vol. 76, no. 1, pp. 141–157, 1996.

[24] T. Bertoni, E. Magosso, and A. Serino, "From statistical regularities in multisensory inputs to peripersonal space representation and body ownership: Insights from a neural network model," *European Journal of Neuroscience*, vol. 53, no. 2, pp. 611–636, 2021.

[25] R. J. Bufacchi, M. Liang, L. D. Griffin, and G. D. Iannetti, "A geometric model of defensive peripersonal space," *Journal of Neurophysiology*, vol. 115, no. 1, pp. 218–225, 2016.

[26] C. F. Sambo, M. Liang, G. Cruccu, and G. D. Iannetti, "Defensive peripersonal space: the blink reflex evoked by hand stimulation is increased when the hand is near the face," *Journal of neurophysiology*, vol. 107, no. 3, pp. 880–889, 2012.

[27] A. Roncone, M. Hoffmann, U. Pattacini, L. Fadiga, and G. Metta, "Peripersonal space and margin of safety around the body: learning visuo-tactile associations in a humanoid robot with artificial skin," *PloS ONE*, vol. 11, no. 10, p. e0163713, 2016.

[28] G. Metta, L. Natale, F. Nori, G. Sandini, D. Vernon, L. Fadiga, C. Von Hofsten, K. Rosander, M. Lopes, J. Santos-Victor *et al.*, "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] R. J. Bufacchi, R. Somervail, A. M. Fitzpatrick, R. Caminiti, and G. D. Iannetti, "Egocentric value maps of the near-body environment," *bioRxiv*, pp. 2022–08, 2022.

[30] T. Kohonen, "The self-organizing map," *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1464–1480, 1990.

[31] 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 Transactions on Cognitive and Developmental Systems*, vol. 10, no. 2, pp. 163–176, 2017.

[32] J. G. Makin, M. R. Fellows, and P. N. Sabes, "Learning multisensory integration and coordinate transformation via density estimation," *PLoS Computational Biology*, vol. 9, no. 4, p. e1003035, 2013.

[33] Z. Straka and M. Hoffmann, "Learning a peripersonal space representation as a visuo-tactile prediction task," in *Artificial Neural Networks and Machine Learning–ICANN 2017: 26th International Conference on Artificial Neural Networks, Alghero, Italy, September 11-14, 2017, Proceedings, Part I 26*. Springer, 2017, pp. 101–109.

[34] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 743–761, 2012.

[35] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," in *CVPR*, 2009.

[36] Z. Straka, T. Svoboda, and M. Hoffmann, "Precnet: Next-frame video prediction based on predictive coding," *IEEE Transactions on Neural Networks and Learning Systems*, 2023, [in press].

[37] R. P. Rao and D. H. Ballard, "Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects," *Nature neuroscience*, vol. 2, no. 1, pp. 79–87, 1999.

[38] A. Knoll, M.-O. Gewaltig, J. Sanders, and J. Oberst, "Neurorobotics: A strategic pillar of the human brain project," *Science Robotics*, pp. 2–3, 2016.

[39] J. Geukes, M. Nakatenus, and R. Calandra, "icub-gazebo-skin," 2017. [Online]. Available: https://github.com/robertocalandra/icub-gazebo-skin

[40] A. S. Leyton and C. S. Sherrington, "Observations on the excitable cortex of the chimpanzee, orang-utan, and gorilla," *Quarterly Journal of Experimental Physiology: Translation and Integration*, vol. 11, no. 2, pp. 135–222, 1917.

[41] W. Penfield and E. Boldrey, "Somatic motor and sensory representation in the cerebral cortex of man as studied by electrical stimulation," *Brain*, vol. 60, no. 4, pp. 389–443, 1937.

[42] 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.

[43] K. Malinovská, I. Farkaš, J. Harvanová, and M. Hoffmann, "A connectionist model of associating proprioceptive and tactile modalities in a humanoid robot," in *2022 IEEE International Conference on Development and Learning (ICDL)*, 2022, pp. 336–342.

[44] F. Gama and M. Hoffmann, "The homunculus for proprioception: Toward learning the representation of a humanoid robot's joint space using self-organizing maps," in *Proceedings of the 2019 Joint IEEE 9th International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob), Oslo, Norway, August 19-22, 2019*. IEEE, 2019, pp. 113–114.

[45] W. J. Ma, J. M. Beck, P. E. Latham, and A. Pouget, "Bayesian inference with probabilistic population codes," *Nature neuroscience*, vol. 9, no. 11, pp. 1432–1438, 2006.

[46] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[47] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2636–2645.

[48] A. Pitonak, "Application of predictive coding for visuo-tactile sensory integration," Bachelor Thesis, CTU in Prague, 2020.

[49] I. Ronga, M. Galigani, V. Bruno, J.-P. Noel, A. Gazzin, C. Perathoner, A. Serino, and F. Garbarini, "Spatial tuning of electrophysiological responses to multisensory stimuli reveals a primitive coding of the body boundaries in newborns," *Proceedings of the National Academy of Sciences*, vol. 118, no. 12, p. e2024548118, 2021.

Appendix A

Robotic homunculus: Learning of artificial skin representation in a humanoid robot motivated by primary somatosensory cortex

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).

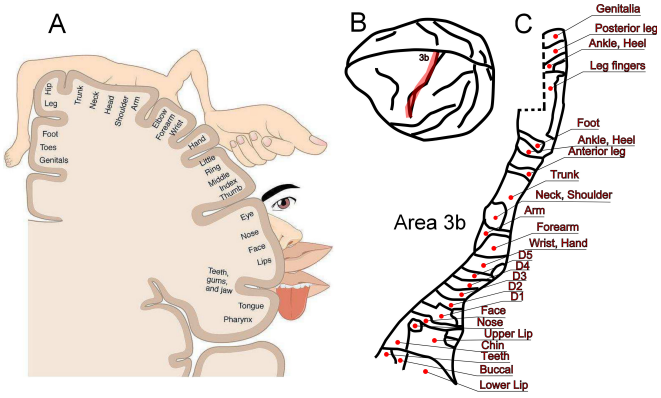

Fig. 1. **Somatosensory homunculus.** (A) Famous somatosensory homunculus of man after Penfield and Rasmusen [3]. Reprinted from [5] under a CC BY license, with permission from OpenStax College, original copyright 2013. Download for free at http://cnx.org/contents/29cade27-ba23-4f4a-8cbd-128e72420f31@5. (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

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.
Manuscript received ...

number of factors, including the presence of neural activity in 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.

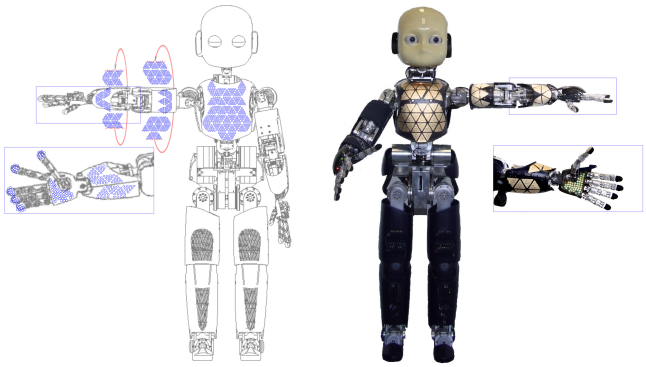

Fig. 2. **Artificial skin on the iCub.** (left) Schematic illustration of the layout of skin patches on one half of the upper body. The patches covering arm and forearm that would not be visible in this view have been unfolded. (right) Photograph of the real robot in analogous posture and exposed skin 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.

$$S(i,j) = \begin{cases} 1 & \text{for a stimulated taxel} \\ 0 & \text{for a non-stimulated taxel} \end{cases} \quad (1)$$

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

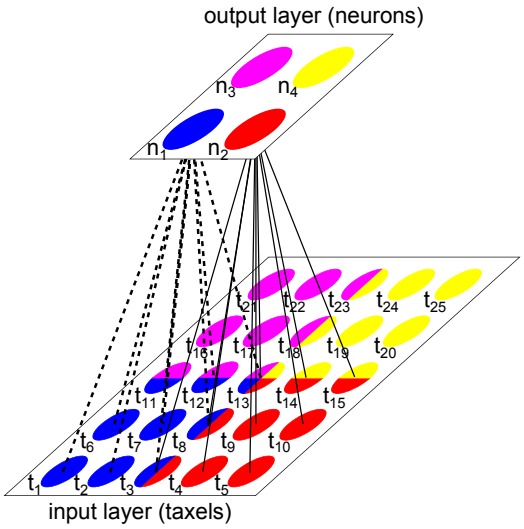

Fig. 3. **Illustration of MRF-SOM.** Four output neurons, n_1, n_2, n_3, n_4, are shown at the top. At the bottom layer, there are 25 inputs, taxels $t_1, ..., t_{25}$. The color code and the weight vectors (weights shown only for $n1, n2$) mark the maximum receptive field size setting of the output neurons. See text for details.

"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.

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:

$$\mathbf{mask_1} = [1,1,1,0,0,1,1,1,0,0,1,1,1,0,0, \\ 0,0,0,0,0,0,0,0,0,0] \quad (2)$$

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].

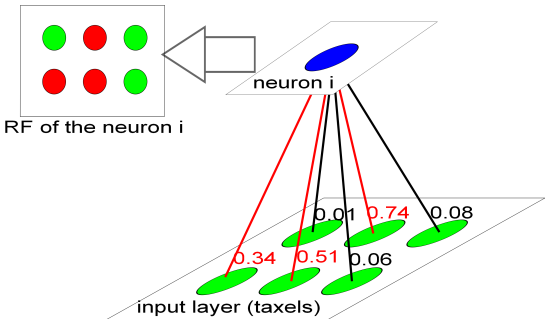

Fig. 4. **Receptive field determination from weight vectors.** For a given threshold, here 0.3, all taxels connected with a neuron with a weight exceeding the chosen threshold are marked as belonging to the RF of the neuron.

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.

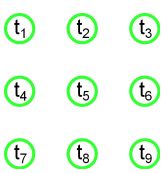

Fig. 5. Schematic illustration of a 3×3 skin patch with 9 taxels.

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

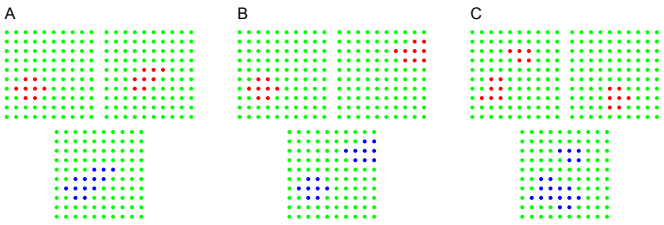

Fig. 6. **Union of two receptive fields.** Top: In each case (A, B, C), the red taxels in the two panels represent hypothetical RFs of two adjacent neurons. Bottom: Blue taxels represent the union of the RFs from the top row. The distances between every taxel pair in this union forms the basis of the 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.

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.

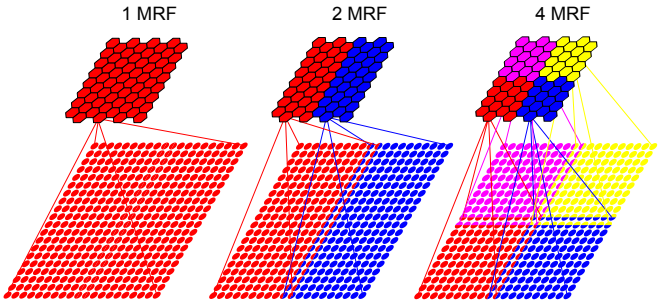

Fig. 7. **Illustration of three variants of MRF setting for simulated skin experiments.** From left to right: 1 MRF, 2 MRF, 4 MRF. There are 8×8 output neurons shown at the top and 20×20 inputs (simulated taxels) at the bottom. The color code and the span of weight vectors mark the maximum receptive field size of every output neuron area. Taxels with multiple colors mark the overlap of maximum receptive fields.

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

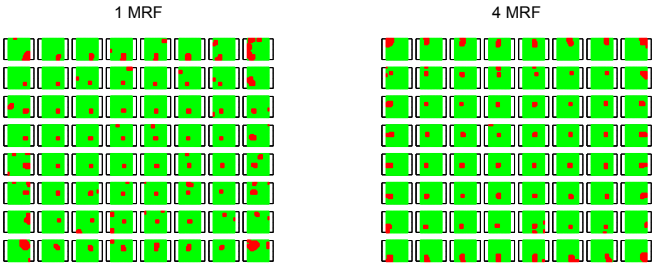

Fig. 8. **Results – Learning from 8-touches in simulated skin. 1 MRF (left) vs. 4 MRF (right) settings.** The 8×8 matrix is the lattice of the output neurons. Every element (subplot) then depicts a miniature version of the simulated skin, in which the set of red taxels represents the receptive field of the corresponding neuron. If there is more than one red area per subplot, it means the neuron's RF is not continuous.

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 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

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

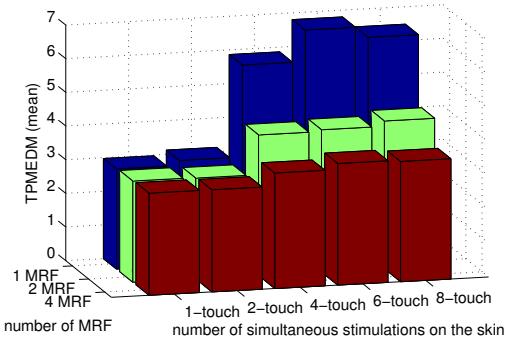

Fig. 9. **Results – Multi-touch on simulated skin – graphical representation of the means from Table II** Lower values correspond to better maps in terms of TPMEDM. The plot reveals the drop in performance of standard SOM (∼ 1 MRF) when faced multi-touch and how this effect is counterbalanced by the use of MRF.

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.

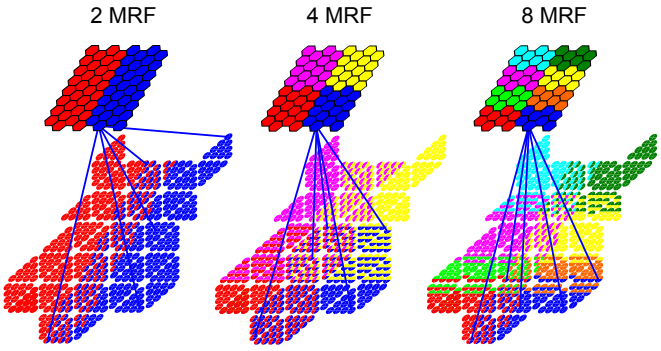

Fig. 10. **Illustration of three variants of the MRF setting for multi-touch on iCub torso.** From left to right: 2 MRF, 4 MRF, 8 MRF (1 MRF not shown). There are 8×8 output neurons at the top; 440 taxels of torso skin in a 2D arrangement (before its attachment on the 3D robot torso) are at the bottom. 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

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

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.

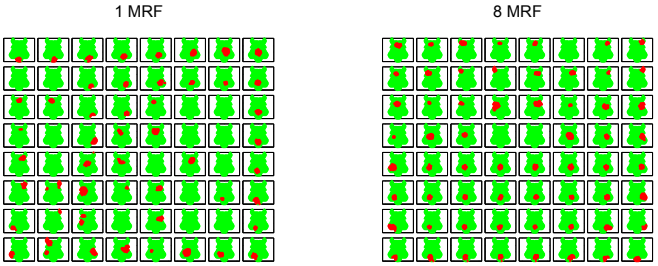

Fig. 11. **Results – Learning from 2-touches in iCub torso skin. 1 MRF (left) vs. 8 MRF (right) settings.** Every element of the 8×8 matrix depicts a miniature version of the torso skin array, in which the set of red taxels represents RF of the corresponding neuron—according to the position on the lattice.

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 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

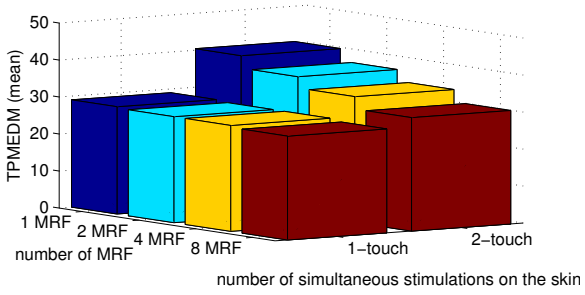

Fig. 12. **Results – Single touch and 2-touch on iCub torso skin – graphical representation of the means from Table III**. Lower values correspond to better maps in terms of TPMEDM. The plot reveals the drop in performance on 2-touch and how this effect is counterbalanced by the use of MRF.

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.

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).

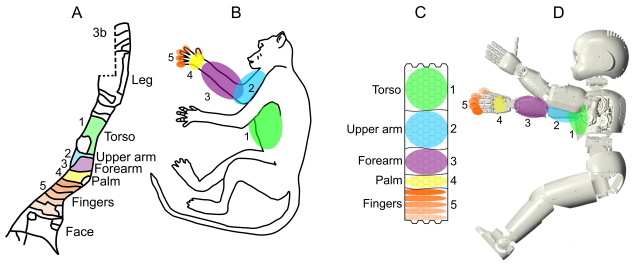

Fig. 13. **Representation of tactile body surface in monkey and robot.** (A-B) Simplified representation of selected body parts in area 3b of macaque monkey. Numbers and color code mark the correspondences between the cortical areas and skin surface on the body parts that will be modeled using the iCub robot. Redrawn and adapted after [4]. (C-D) Schematics of analogous situation in the robot – approximate target for the SOM algorithm.

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 (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.

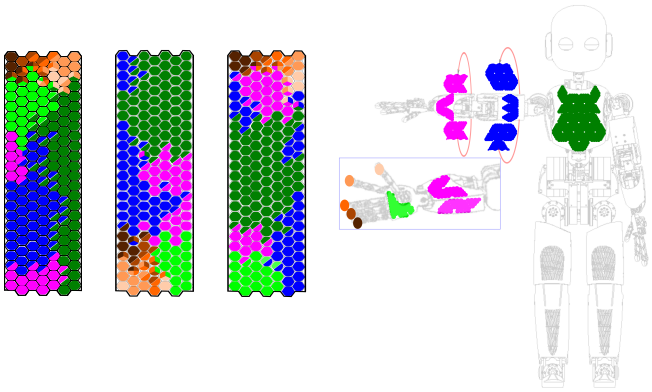

Fig. 14. **Results – Learning from tactile stimulation on right side of robot body with standard DP-SOM.** The three panels on the left depict the maps (24×7 neurons) after learning as a result of three runs of the algorithm on the same training set. The visualization, which colors the maps according 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.

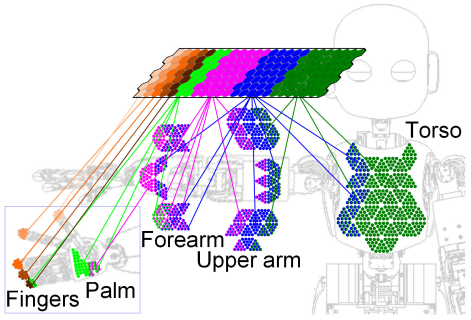

Fig. 15. **Detailed MRF setting for learning tactile homunculus.** The colors and lines ascending from the individual skin parts to the neuronal sheet schematically illustrate the MRF settings. Every skin part has its "dedicated" area in the neuronal sheet—dark green for torso, blue for upper arm, pink 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.

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 preservation 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.

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

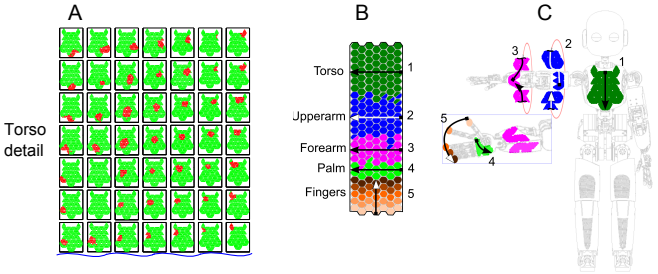

Fig. 16. **Results – Learning from tactile stimulation with MRF-SOM.** (A) Top section of the output map – 7×7 neurons with miniature depictions of the torso skin; red taxels mark RFs of the corresponding neuron (visualization using the "biomimetic RF determination method"; see Section II-D). (B) Visualization 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].

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].

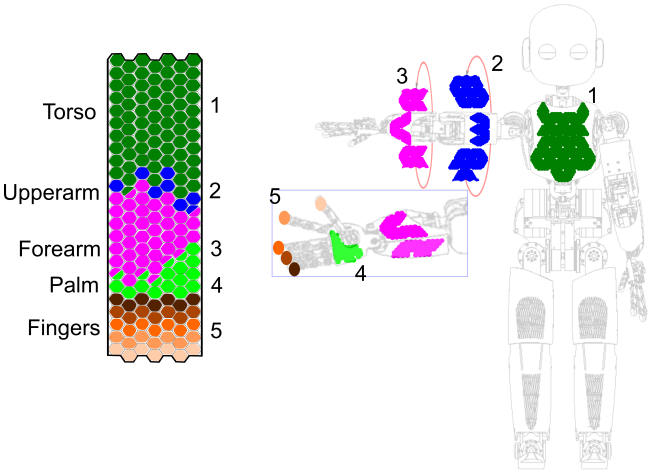

Fig. 17. **Results – Learning from tactile stimulation with MRF-SOM and simulated lesion of upper arm.** The same settings and visualization as in Fig. 16 were used, but 3/4 of upper arm stimulations were pruned in the training set.

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 B

Learning a peripersonal space representation as a visuo-tactile prediction task

The final publication [33] is available at Springer via `https://doi.org/10.1007/978-3-319-68600-4_13`.

Learning a Peripersonal Space Representation as a Visuo-Tactile Prediction Task

Zdenek Straka[1( )] and Matej Hoffmann[1,2]

[1] Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, Prague, Czech Republic
{straka.zdenek,matej.hoffmann}@fel.cvut.cz
[2] iCub Facility, Istituto Italiano di Tecnologia, Genoa, Italy

Abstract. The space immediately surrounding our body, or peripersonal space, is crucial for interaction with the environment. In primate brains, specific neural circuitry is responsible for its encoding. An important component is a safety margin around the body that draws on visuo-tactile interactions: approaching stimuli are registered by vision and processed, producing anticipation or prediction of contact in the tactile modality. The mechanisms of this representation and its development are not understood. We propose a computational model that addresses this: 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). Unique to this model, it considers: (i) stimulus position and velocity, (ii) uncertainty of all variables, and (iii) not only multisensory integration but also prediction.

Keywords: Peripersonal space · Touch · RBM · Probabilistic population code · Visuo-tactile integration

1 Introduction

For survival, animals and humans have to be "aware" of their bodies and space around them. This space is called peripersonal space (PPS) and is especially important for safe interaction of an agent with the environment. PPS is the space that extends the surface of the body. In the primate brain, there is neural circuitry specialized on PPS representation, in particular bimodal neurons with visuo-tactile receptive fields (e.g., [3]; [1] for a review) firing when some part of the skin is stimulated or a visual stimulus is presented nearby. The PPS is seemingly extended when a stimulus moves faster (e.g., [3]) and the direction of the moving object (looming vs. receding) is also important for responses of the PPS network [10]. Thus, position and velocity of the stimulus have to be considered. Moreover, there is evidence that the brain is able to combine different sensory information in a statistically optimal manner ([2]; [5] for a computational model), for which the brain must also encode uncertainty of sensory information. The two modalities—visual and tactile—are presumably interacting in several

A. Lintas et al. (Eds.): ICANN 2017, Part I, LNCS 10613, pp. 101–109, 2017.
https://doi.org/10.1007/978-3-319-68600-4_13

ways: (i) the correlations induced when the stimulus contacts the skin surface may facilitate learning and online adaptation of the PPS; (ii) the visual information is predictive of the tactile in both space and time—that is, an approaching stimulus that is perceived only visually facilitates the responses of neurons with tactile receptive fields at the expected contact location (e.g., [10]).

PPS learning—in a narrow sense of the visuo-tactile neurons' characteristics—can be viewed as a regression task: learning a functional relationship between a visual stimulus in space (position and velocity) and the expected contact location as perceived by the tactile modality. Training data is provided by approaching objects perceived visually and eventually contacting the skin. If uncertainty of the input is considered, we obtain a regression problem with errors in variables.

There are few computational models of PPS representation learning in the sense considered here (i.e., PPS as margin-of-safety rather than PPS as space within reach – see [1]). Magosso et al. [6] proposed a neural network that models unimodal (visual and tactile) and bimodal representations of an imaginary left and right body part, but focused on their interaction rather than learning and velocity was not considered. Roncone et al. [9], on a humanoid robot, developed a proxy for the visual receptive fields in a probabilistic sense (likelihood of contact) and showed that they can be learned from scratch from objects nearing and eventually contacting the skin. Velocity (or time to contact) was considered, but for both, position and velocity, the 3D space was collapsed to a single dimension. Neither of the models takes uncertainty of the inputs into account.

Our work departs from a neural network model based on a Restricted Boltzmann Machine (RBM) from [7] that enables integration of information from different modalities—there vision and proprioception, here position and velocity both derived from vision (the step of extracting these quantities from actual visual input is not addressed here). A probabilistic population code [5] is used to encode position and velocity as Gaussian distributions including uncertainty, which are then fed into the RBM model providing dimensionality reduction and feature extraction. However, the model is not able to make temporal predictions such as predicting the future state of one modality from the other modality. Thus, we extended the model by a feedforward neural network that takes the RBM hidden neurons as input and learns to predict a location on the body surface (covered by skin) that will be hit by a moving object based on the integrated representation of position and velocity of the object.

This article is structured as follows. The Materials and Methods section details input/output encoding and the RBM. This is followed by the Experiments and Results section where we describe learning and testing of the model. We close with a Conclusion and Discussion.

2 Materials and Methods

2.1 Input and Output Encoding

The input neurons use a "probabilistic population code" [5,7] to encode a measurement $\mathbf{x}$ and its uncertainty (determined by a gain g). A state (or "activation") $\mathbf{r}$ of the neuron population is sampled from the distribution

$$p(\mathbf{r}|\mathbf{x}, g, \mathbf{\Sigma}_t) = \prod_j Pois[r_j|gf_j(\mathbf{x})], \tag{1}$$

where $f_j(\mathbf{x}) = e^{-\frac{1}{2}(\mathbf{x}-\mathbf{c}_j)^T \mathbf{\Sigma}_t (\mathbf{x}-\mathbf{c}_j)}$ is a Gaussian function centered in the receptive field (RF) center $\mathbf{c}_j$ of the j-th neuron, the covariance matrix $\mathbf{\Sigma}_t$ is a constant diagonal matrix (for the given modality), with all diagonal elements having the same value (variance of the Gaussian function) that determines the width of the RF.

The state $\mathbf{r}$ of the neuron population can be interpreted as a normal distribution (we assume that the size of the neuron population is sufficiently large) $\mathcal{N}(\psi(\mathbf{r}), \overline{\mathbf{\Sigma}}(\mathbf{r}))$ [5,7] where

$$\psi(\mathbf{r}) = \frac{\sum_i \mathbf{c}_i r_i}{\sum_i r_i} \tag{2}$$

is the mean and

$$\overline{\mathbf{\Sigma}}(\mathbf{r}) = \frac{\mathbf{\Sigma}_t}{\sum_i r_i} \tag{3}$$

is the covariance matrix. The matrix is diagonal, with all diagonal elements equal to the variance σ^2. Equations (2) and (3) are valid if we assume that a prior distribution $p(\mathbf{x})$ is uniform (for the Gaussian case see [7]). A relationship between g and the variance is $g \propto \frac{1}{\sigma^2}$ [5]. In what follows, instead of the covariance matrix (3), we will use η and call it *confidence* of a measurement, defined as follows:

$$\eta = \sum_i r_i \tag{4}$$

The confidence η fully determines the values of the covariance matrix (see the denominator in (3)). Note that $\eta \propto g \propto \frac{1}{\sigma^2}$. Thus, the decoded covariance σ^2 as the uncertainty of the measurement can always be determined from η.

For detailed information about neuron RF centers $\mathbf{c}_j^{pos}$, $\mathbf{c}_j^{vel}$, c_j^{tact} see [11].

2.2 Restricted Boltzmann Machine (RBM)

This part of the architecture is based on an RBM-like model from [7]. A Restricted Boltzmann machine is a generative model that consists of two layers with no intralayer connections and full interlayer connections [4,12] (see Fig. 1 right). The input units (with state $\mathbf{r}$) are Poisson random variables that take nonnegative integer values according to (1). The hidden-layer units (with state $\mathbf{v}$) are binary. The input and hidden units have biases ($\mathbf{b}_r$, $\mathbf{b}_v$). The connection between both layers (weights $\mathbf{W}$) is undirected.

During learning, one population is given and the other is sampled. The units $\mathbf{v}$ (resp. $\mathbf{r}$) are sampled from Bernoulli (Poisson) distribution [4,12]

$$p(\mathbf{v}|\mathbf{r}) = \prod_i Bern[v_i|\sigma(\{\mathbf{W}\mathbf{r} + \mathbf{b}_v\}_i)] \tag{5}$$

$$p(\mathbf{r}|\mathbf{v}) = \prod_j Pois[r_j|exp(\{\mathbf{W}^\mathbf{T}\mathbf{v} + \mathbf{b}_r\}_j)] \quad (6)$$

The RBM was trained using one-step contrastive divergence [4].

3 Experiments and Results

We deploy our neural network architecture in a 2D scenario where objects are approaching a simulated skin surface (see Fig. 1 left). The performance of the learned representation is assessed, focusing on the precision and reliability of the predictions generated. Finally, we analyze how the PPS representation is modulated by stimulus speed. Complete code and parameters for all experiments is available online [11].

3.1 Peripersonal Space Representation Learning

Learning proceeds in two separate phases. The input variables are: (i) 2D stimulus position, $\mathbf{x}^{pos}$ (from the hypothetical "visual" modality), and (ii) stimulus velocity, $\mathbf{x}^{vel}$ – the change of position during a timestep $\mathbf{x}^{vel}(t) = \mathbf{x}^{pos}(t) - \mathbf{x}^{pos}(t-1)$. Both are encoded (using (1)) by the neural populations with states $\mathbf{r}^{pos}$ and $\mathbf{r}^{vel}$ respectively. The gains associated with the input variables, g^{pos}, g^{vel}, are uniformly generated from bounded intervals. First, the RBM is trained to represent this input space in an unsupervised fashion. Second, the tactile modality is added and learning proceeds in a supervised way to predict the contact location.

RBM Learning. The object positions $\mathbf{x}^{pos}(i), i \in \{1, 2, ..., N\}$ (N is the size of the training set) uniformly covered the space of the visual modality (see Fig. 1 left). The direction and magnitude of each velocity vector

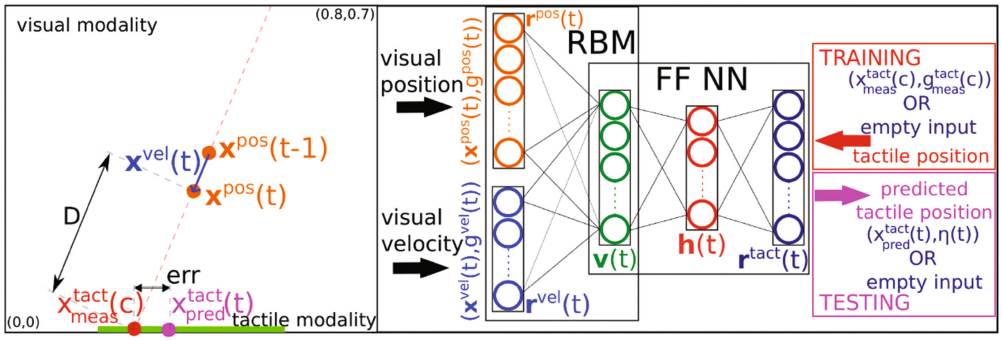

Fig. 1. Scenario and architecture. LEFT: 2D experimental scenario. Stimulus trajectory in orange; positions of stimulus at two different discrete time moments shown. "Skin" in green. **RIGHT:** Architecture of the neural network and illustration of training and testing (predicting) process. See text for details. (Color figure online)

$\mathbf{x}^{vel}(i), i \in \{1, 2, ..., N\}$ were uniformly generated from a bounded interval. For training of the RBM, we used the training set $U = \{[\mathbf{r}_U^{pos}(1), \mathbf{r}_U^{vel}(1)], [\mathbf{r}_U^{pos}(2), \mathbf{r}_U^{vel}(2)], ..., [\mathbf{r}_U^{pos}(N), \mathbf{r}_U^{vel}(N)]\}$, where $\mathbf{r}_U^{pos}(i)$ and $\mathbf{r}_U^{vel}(i)$ are obtained from $\mathbf{x}^{pos}(i)$ and $\mathbf{x}^{vel}(i)$ (using (1)). The RBM was trained using one-step contrastive divergence [4]. The main parameters of the learning were: $size(\mathbf{r}^{pos}) = 289$, $size(\mathbf{r}^{vel}) = 625$, $size(\mathbf{v}) = 150$, $g^{pos/vel} \in (12, 18)$, $\mathbf{x}^{vel} \in (-0.012, 0.012) \times (-0.012, 0.012)$ and the number of training epochs was 60 (for other parameters see [11]).

Feedforward Network Learning. The second phase of learning can be viewed as a regression task, with $\mathbf{x}^{pos}$ and $\mathbf{x}^{vel}$ as independent variables and x^{tact}, 1D position of the stimulation registered by the tactile modality, as the dependent variable (can be empty – no prediction). As before, all variables have their respective gains $g^{pos}, g^{vel}, g^{tact}$ (uniformly generated from bounded intervals) and are encoded using (1), giving $\mathbf{r}^{pos}$, $\mathbf{r}^{vel}$, and $\mathbf{r}^{tact}$. We will distinguish predicted value of the tactile position x_{pred}^{tact} and the measured value x_{meas}^{tact} that are used during training and testing.

Simulated looming objects follow trajectories that start at the top edge of a simulated space (dimensions chosen arbitrarily) and end at the bottom edge (see Fig. 1 left). The start and end of the trajectory and the object velocity are generated uniformly from bounded intervals. If the end of the trajectory falls in the region covered by the emulated "skin", the tactile modality is activated. The position of the stimulation object is recorded at discrete time moments (see the orange circles in Fig. 1 left).

The relationship between "visual" stimulation, $e(t)$, and tactile stimulation, $z(t)$, is formally described below. On contact of the object with "skin", the "connection" is strengthened if the tactile stimulation $x_{meas}^{tact}(c)$ follows the moment of "visual" stimulation at time t by at most Q timesteps. Formally, let $C \subset \{1, 2, .., M\}$ be the set of time moments when the tactile modality was activated, M size of the training set and Q an integer constant ("memory buffer size"). The set T consists of pairs $T = \{(e(1), z(1)), (e(2), z(2)), ..., (e(M), z(M))\}$, where $e(t) = (\mathbf{x}^{pos}(t), g^{pos}(t), \mathbf{x}^{vel}(t), g^{vel}(t))$ (independent variables with their gains) and $z(t) = (x_{meas}^{tact}(c), g_{meas}^{tact}(c))$ if $\exists c, c \in C$ that $t \in [c - Q, c]$, else $z(t)$ is empty.

For training of a feedforward neural network (FF NN), the set T will now be used to generate a set $S = \{(\mathbf{v}(1), \mathbf{r}^{tact}(1)), (\mathbf{v}(2), \mathbf{r}^{tact}(2)), ..., (\mathbf{v}(M), \mathbf{r}^{tact}(M))\}$, where $\mathbf{v}$ is the state of the RBM hidden layer and is sampled from the Bernoulli distribution (5) given $\mathbf{r} = [\mathbf{r}^{pos}, \mathbf{r}^{vel}]$, as obtained from $e(t)$. Then, $\mathbf{r}^{tact}(t)$ is obtained from a corresponding $z(t)$ – see Fig. 1 right. If $z(t)$ is empty, then $\mathbf{r}^{tact}(t)$ is a zero vector.

We used a standard two-layer feedforward neural network with sigmoid hidden neurons (state denoted $\mathbf{h}$) and linear output neurons (see Fig. 1 right). The training algorithm was scaled conjugate gradient backpropagation [8]. For the training we used MATLAB's Neural Network Toolbox. The main parameters of the learning were: $size(\mathbf{r}^{tact}) = 25$, $size(\mathbf{h}) = 20$, $Q = 70$, $g^{pos/vel/tact} \in (12, 18)$, $||\mathbf{x}^{vel}|| \in (0.005, 0.01)$ and the number of training epochs was 3369.

3.2 Peripersonal Space Representation Testing

The process of prediction is schematically illustrated in Fig. 1 right. The prediction is obtained from the feedforward neural network. An input $\mathbf{v}$ of the FF NN is obtained from the stimulus $(\mathbf{x}^{pos}, \mathbf{x}^{vel}, g^{pos}, g^{vel})$ in the same way as it is described in Sect. 3.1. From the output of the FF NN $\mathbf{r}^{tact}$ (to prevent negative activations and noise, we set to zero all r_j^{tact} that have smaller value than 1), we can get the predicted position $x_{pred}^{tact}(i) = \psi(\mathbf{r}^{tact}(i))$ and the confidence $\eta(i)$ (see Eqs. (2) and (4)). If all elements of a state $\mathbf{r}^{tact}$ are zeros, then no prediction is generated. The error of the prediction is $err = |x_{pred}^{tact} - x_{meas}^{tact}|$ (see Fig. 1 left). For testing we use x_{meas}^{tact} for the end point of the trajectory (even if it lies outside of the space covered by skin – cannot be "measured" by the tactile

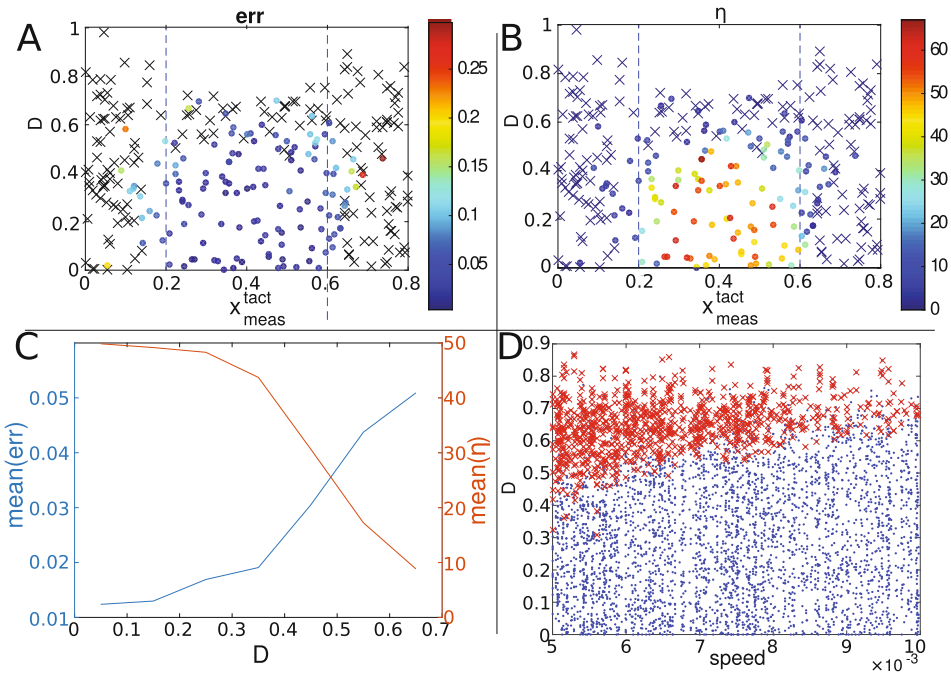

Fig. 2. Peripersonal space representation testing – touch prediction performance. A: Dependence of error on distance D and end of the trajectory $\mathbf{x}_{\mathbf{meas}}^{\mathbf{tact}}$. The color code encodes the error $|x_{meas}^{tact} - x_{pred}^{tact}|$ in actual vs. predicted contact location (for the meaning of the D, x_{meas}^{tact} and err see Fig. 1). The crosses denote that the prediction is not generated ($\mathbf{r}^{tact} = \mathbf{0}$). The area between the two dashed lines contains the stimuli that are followed by the tactile stimulation (x_{meas}^{tact} is on the skin). **B: Dependence of confidence on distance D and end of the trajectory $\mathbf{x}_{\mathbf{meas}}^{\mathbf{tact}}$.** The color code encodes the confidence (see (4)) of each prediction depending on D and x_{meas}^{tact}. **C: Dependence of error and confidence on D.** Only trajectories with end on the skin were used (the area between the dotted lines in A, B). Each value of the dependent variable is the mean of error or confidence for stimuli from a 0.1 wide area of D. Empty predictions were excluded. **D: Dependence of prediction on speed and distance D.** Each point represents a moving stimulus (with a known value of speed) at distance D from the end of trajectory. All stimuli were from trajectories that end on the skin. The stimuli for which predictions are empty are marked by a red cross, others are marked by a blue dot. (Color figure online)

modality). The stimuli for testing are obtained in the same way as for learning (see Sect. 3.1). The results are analyzed in the next section.

3.3 Analysis of the Results

The results are summarized in Fig. 2. Overall, the architecture has successfully coped with the task. We find that if the trajectory of the stimulus ends on the skin, the prediction error increases with the distance from the contact location, but the prediction confidence decreases. This is illustrated in aggregated form in Fig. 2C and in detail in Fig. 2A, B (in the latter, the testing set is reduced for visualization purposes). If the trajectory ends outside the skin ($x_{meas}^{tact} \notin [0.2, 0.6]$), there was no prediction ($\eta = 0$) or the confidence η had a low value (see Fig. 2A, B). This is desirable, as the lower confidence enables recognition of false and inaccurate predictions. It is also possible to see that the confidence was lower at the edges of the skin than in the central part.

In Fig. 2A and B, there seems to be a fuzzy but apparent border or threshold in distance, after which the generated predictions are empty or their confidence is low – around $D = 0.5$. This border is determined by buffer size Q, but, importantly, it is also modulated by speed of the stimulus. We analyzed this specifically in Fig. 2D: with higher speed, the empty predictions are generated farther from the skin, so the "border of the PPS" moves farther.

4 Conclusion and Discussion

The mechanisms of PPS representation and learning in biology are not fully understood. Arguably, PPS adaptation can be largely attributed to neuronal plasticity in the corresponding networks (probably fronto-parietal areas) through interaction with the environment. The contingencies between a visual stimulus looming to the body and tactile stimulation on contact of the object with the skin may constitute sufficient material for the development and continuous recalibration of the PPS representation.

To investigate this hypothesis, we proposed a neural network architecture that consists of two parts. The first network has two input populations, one encodes position of the "visual" stimulus, the other encodes velocity of the stimulus. Both of them also encode uncertainty of the stimuli. The information from the input layers is integrated by the hidden layer of an RBM. However, this model alone cannot make temporal predictions, so we extended it by a feedforward neural network with one hidden layer. This feedforward network is trained in a supervised manner to predict tactile stimulation.

We tested how the network after training can predict tactile stimulation given the "visual" position and velocity of a looming stimulus and found that: (i) the error of the prediction increased with the distance of the stimulus from the skin; (ii) the confidence of the prediction decreased with distance. The confidence was also low or zero if the trajectory of the stimulus ended outside the skin. These are expected and desired properties, thus verifying the suitability of our method.

Interestingly, our model reproduced the phenomenon of seeming PPS expansion pertaining to faster stimuli and predicts a hypothetical mechanism for this: for a given distance, there is an emergent cut-off speed, whereby slower stimuli do not induce any prediction of touch (and thus may not lead to PPS activation) but faster stimuli do.

In the future, we want to conduct a more detailed comparison with the properties of PPS in biology. In addition, it will be natural to add additional modalities next to vision and touch: (i) the auditory modality may provide additional information about the same stimulus, which in turn needs to be optimally integrated with vision; (ii) proprioception is mediating coordinate transformations for stimuli pertaining to the body. Finally, we want to test our model in a real scenario on a humanoid robot. These may require changes to the architecture presented here, such as possible recruitment of a convolutional neural network to process raw visual inputs, and—upon inclusion of additional modalities and hence dimensions to the task—transforming the RBM into a Deep belief network or adding more hidden layers to the FF NN.

Acknowledgement. Z.S. was supported by The Grant Agency of the CTU Prague project SGS16/161/ OHK3/2T/13. M.H. was supported by the Czech Science Foundation under Project GA17-15697Y and a Marie Curie Intra European Fellowship (iCub Body Schema 625727) within the 7th European Community Framework Programme. Base code for the RBM model was kindly provided by Joseph G. Makin [7].

References

1. Cléry, J., Guipponi, O., Wardak, C., Hamed, S.B.: Neuronal bases of peripersonal and extrapersonal spaces, their plasticity and their dynamics: knowns and unknowns. Neuropsychologia **70**, 313–326 (2015)
2. Ernst, M.O., Banks, M.S.: Humans integrate visual and haptic information in a statistically optimal fashion. Nature **415**(6870), 429–433 (2002)
3. Fogassi, L., Gallese, V., Fadiga, L., Luppino, G., Matelli, M., Rizzolatti, G.: Coding of peripersonal space in inferior premotor cortex (area f4). J. Neurophysiol. **76**(1), 141–157 (1996)
4. Hinton, G.E.: Training products of experts by minimizing contrastive divergence. Neural Comput. **14**(8), 1771–1800 (2002)
5. Ma, W.J., Beck, J.M., Latham, P.E., Pouget, A.: Bayesian inference with probabilistic population codes. Nat. Neurosci. **9**(11), 1432–1438 (2006)
6. Magosso, E., Zavaglia, M., Serino, A., Di Pellegrino, G., Ursino, M.: Visuotactile representation of peripersonal space: a neural network study. Neural Comput. **22**(1), 190–243 (2010)
7. Makin, J.G., Fellows, M.R., Sabes, P.N.: Learning multisensory integration and coordinate transformation via density estimation. PLoS Comput. Biol. **9**(4), e1003035 (2013)
8. Møller, M.F.: A scaled conjugate gradient algorithm for fast supervised learning. Neural Netw. **6**(4), 525–533 (1993)
9. Roncone, A., Hoffmann, M., Pattacini, U., Fadiga, L., Metta, G.: 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 (2016)

10. Serino, A., Noel, J.P., Galli, G., Canzoneri, E., Marmaroli, P., Lissek, H., Blanke, O.: Body part-centered and full body-centered peripersonal space representations. Sci. Rep. **5**, 18603 (2015)
11. Straka, Z., Hoffmann, M.: Supporting materials. https://github.com/ZdenekStraka/icann2017-pps
12. Welling, M., Rosen-Zvi, M., Hinton, G.E.: Exponential family harmoniums with an application to information retrieval. In: Proceedings of NIPS, vol. 4, pp. 1481–1488 (2004)

Appendix C

A normative model of peripersonal space encoding as performing impact prediction

PLOS COMPUTATIONAL BIOLOGY

RESEARCH ARTICLE

A normative model of peripersonal space encoding as performing impact prediction

Zdenek Straka[1], **Jean-Paul Noel**[2], **Matej Hoffmann**[1]*

1 Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, Prague, Czech Republic, **2** Center for Neural Science, New York University, New York City, New York, United States of America

* matej.hoffmann@fel.cvut.cz

OPEN ACCESS

Citation: Straka Z, Noel J-P, Hoffmann M (2022) A normative model of peripersonal space encoding as performing impact prediction. PLoS Comput Biol 18(9): e1010464. https://doi.org/10.1371/journal.pcbi.1010464

Editor: Arvid Guterstam, Karolinska Institutet, SWEDEN

Received: February 11, 2022

Accepted: August 2, 2022

Published: September 14, 2022

Data Availability Statement: The Matlab code is available at https://github.com/ctu-vras/pps-normative-model.

Funding: This study was supported by Czech Science Foundation (GA CR, https://gacr.cz/en/), project no. 20-24186X awarded to M.H. Z.S. and M.H. were supported by this project. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

Abstract

Accurately predicting contact between our bodies and environmental objects is paramount to our evolutionary survival. It has been hypothesized that multisensory neurons responding both to touch on the body, and to auditory or visual stimuli occurring near them—thus delineating our peripersonal space (PPS)—may be a critical player in this computation. However, we lack a normative account (i.e., a model specifying how we *ought to* compute) linking impact prediction and PPS encoding. Here, we leverage Bayesian Decision Theory to develop such a model and show that it recapitulates many of the characteristics of PPS. Namely, a normative model of impact prediction (i) delineates a graded boundary between near and far space, (ii) demonstrates an enlargement of PPS as the speed of incoming stimuli increases, (iii) shows stronger contact prediction for looming than receding stimuli—but critically is still present for receding stimuli when observation uncertainty is non-zero—, (iv) scales with the value we attribute to environmental objects, and finally (v) can account for the differing sizes of PPS for different body parts. Together, these modeling results support the conjecture that PPS reflects the computation of impact prediction, and make a number of testable predictions for future empirical studies.

Author summary

The brain has neurons that respond to touch on the body, as well as to auditory or visual stimuli occurring near the body. These neurons delineate a graded boundary between the near and far space. Here, we aim at understanding whether the function of these neurons is to predict future impact between the environment and body. To do so, we build a mathematical model that is statistically optimal at predicting future impact, taking into account the costs incurred by an impending collision. Then we examine if its properties are similar to those of the above-mentioned neurons. We find that the model (i) differentiates between the near and far space in a graded fashion, predicts different near/far boundary depths for different (ii) body parts, (iii) object speeds and (iv) directions, and (v) that this boundary scales with the value we attribute to environmental objects. These properties have all been described in behavioral studies and ascribed to neurons responding to

objects near the body. Together, these findings suggest *why* the brain has neurons that respond only to objects near the body: to compute predictions of impact.

Introduction

Predicting environmental impact on our body is a critical computation promoting our evolutionary survival. Interactions between our body and the environment occur within the theater of our peripersonal space (PPS; [1, 2]), the space immediately adjacent to and surrounding our body. In turn, the brain has a specialized fronto-parietal circuit representing multisensory objects and events in a body-centered reference frame when these are near the body [3–5]. There is strong experimental evidence demonstrating that PPS plays a key role in defensive behaviors (see [6] for a seminal review) and initial evidence likewise suggests that PPS encoding plays a role in impact prediction [4, 7, 8]. For instance, stimuli looming toward the body enhance tactile sensitivity at the spatial and temporal location where observers expect impact to occur [9], and PPS enlarges as the speed of incoming stimuli grows [10]. However, we lack a normative account linking impact prediction and PPS.

Modeling efforts have accounted for a number of different aspects of PPS. Magosso and colleagues first introduced a biologically motivated neural network of PPS [11, 12]. This model inherits much of its ability to distinguish between near and far spaces from its local connectivity patterns within unisensory areas. Variants of this model can account for PPS re-sizing after tool use [12, 13], as well as its remapping as a function of the speed of approaching stimuli [14] and recent stimuli statistics [15]. This model may also account for the inflexibility of PPS remapping in autism [16]. Similarly, Bertoni et al. [17] developed a neural network model of PPS, with the innovation that this latter one learns the statistical regularities between visual, tactile, and proprioceptive inputs in order to construct a representation of PPS. In doing so, Bertoni et al.'s model shows how PPS neurons may be anchored to body parts. Straka and Hoffmann [18] have trained a neural network to integrate seen object position and velocity, as well as to predict future tactile contact. However, this model's predictions of tactile activation, and thus impact, were trained in a supervised manner and the model did not explicitly calculate the probability of future tactile contact. Roncone et al. [19] proposed a PPS model which was trained using a humanoid robot, by nearing objects. The model estimated the likelihood of future contact and used this prediction for avoidance behavior. Perhaps most related to our model, Bufacchi et al. [20] used a 3D geometric model of defensive PPS to fit hand-blink reflex data, assuming uncertainty about stimulus direction in all 3 dimensions and an infinite time-limit.

These models have certainly advanced our understanding of PPS, but share a common limitation in being non-normative. That is, they suggest how PPS and impact prediction could be computed or learned from observations, as opposed to how it ought to be computed. Instead, a wealth of evidence, across a wide variety of fields and tasks (e.g., [21–24]), have shown that humans perceive and perform decisions (near) optimally. Thus, mechanistic models (e.g., neural networks) and human performance should be benchmarked against statistical optimality. Similarly, a strong test of the hypothesis that a functional role of PPS is to perform impact prediction [4, 8] is to build a normative model of the latter, and then contrast the behavior of this model to known properties of PPS encoding.

Here, we use Bayesian Decision Theory [25–28] to propose a normative model of PPS as performing prediction of impact which minimizes the loss/cost such an impact may incur to the agent. We show that this normative model (i) delineates a graded boundary between near

and far space [3], (ii) demonstrates a larger PPS as the speed of incoming stimuli increases [10, 14], (iii) shows stronger contact prediction for looming than receding stimuli—but critically is still present for receding stimuli [6, 29, 30]—, (iv) scales with the values of objects (e.g., innocuous vs. potentially dangerous; [31, 32]), and finally (v) can account for differing sizes of PPS for different body parts [33]. Together, these results recapitulate a set of important features of PPS and support the hypothesis that PPS neurons perform contact prediction.

Results

We developed a Bayesian observer inferring whether contact between an external object and the body would occur within the next time step. An overview of the model is given in Fig 1 and S1 File (for full detail see the Materials and methods section). Briefly, at time T, an object has position x_T and moves with velocity v_T. The observer is tasked with predicting whether at or before $T + \Delta T$ this object will make contact with the body. This prediction takes into account two components. First, the probability estimation of the object making contact with the body, given its perceived position and velocity, including its uncertainty. Second, the loss (i.e., penalty) incurred if the prediction is incorrect. We denote the possible impact of the object on the body as $y \in \{0, 1\}$, which is a binary variable—either there is contact with the body or there is not. Instead, $y_{pred} \in [0, 1]$, a continuous value, is the prediction whether contact will occur or not, taking into account the estimation of probability of contact and the loss function. Optimal impact prediction is denoted by y^*_{pred}.

According to Bayesian Decision Theory (see e.g., [25, 26]) the optimal decision—in our case the impact prediction y^*_{pred}—is

$$y^*_{pred} = \arg\min_{y_{pred} \in [0,1]} L((\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v), y_{pred}) \tag{1}$$

where

$$\begin{aligned} L((\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v), y_{pred}) = P(y = 1|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v)) \cdot loss(y = 1, y_{pred}) + \\ P(y = 0|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v)) \cdot loss(y = 0, y_{pred}) \end{aligned} \tag{2}$$

and $\hat{x}_T, \hat{v}_T$ are respectively the observer's point estimates of the object position x_T and velocity v_T at time T (see Fig 1). The estimates need not be the same as the actual object position and velocity, given that perception may be distorted by observation noise (see Derivation of the normative impact prediction model for details). Uncertainty about the position and velocity are respectively expressed by σ_x, σ_v. Stimuli perceived less accurately (e.g., visual stimuli at low contrast, or auditory localization as opposed to visual localization) result in greater σ_x and σ_v. To include this uncertainty, position and velocity estimates are respectively encoded as normal distributions $N(\mu = \hat{x}_T, \sigma = \sigma_x)$ and $N(\mu = \hat{v}_T, \sigma = \sigma_v)$. Displacement of the object during ΔT is encoded as normal distribution $N(\mu = \Delta T \cdot \hat{v}_T, \sigma = \Delta T \cdot \sigma_v)$ (see Fig 1 or Derivation of the normative impact prediction model for details).

Merging the position and displacement estimations, the probability $P(y|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v))$ of the external object making contact with the body ($y = 1$) at or before $T + \Delta T$ given the agent's observations at time T is estimated (see the calculation in Fig 1 or in Derivation of the normative impact prediction model). Conversely, the estimated probability that the external object will not make impact with the body is
$P(y = 0|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v)) = 1 - P(y = 1|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v))$.

The second important component in computing the value associated with an object's velocity and distance to the body is the utility function, loss(y, y_{pred}). For a predicted value y_{pred}, it enables to calculate the corresponding loss associated with $y \in \{0, 1\}$. For a zero-one loss

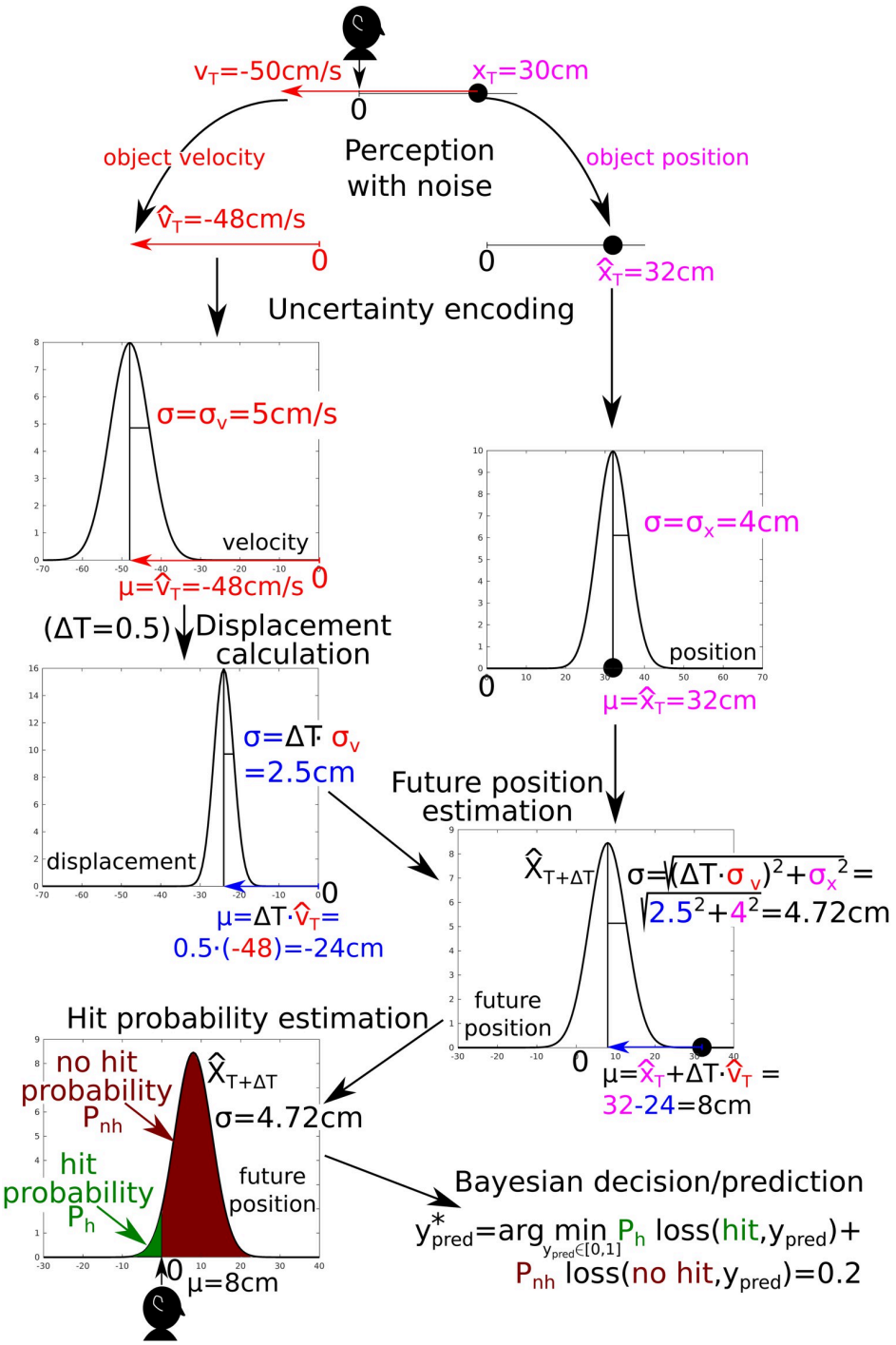

Fig 1. Schema and illustrative example of the contact prediction model. Say an object (black circle) is $x_T = 30$cm from the body (black head) and is approaching with velocity $v_T = -50cm/s$. **Perception with noise**. The nervous system estimates the position and velocity of the object with respect to our body with a given uncertainty. For instance, we may estimate $\hat{x}_T = 32cm$ and $\hat{v}_T = -48cm/s$. Assuming that the noise is Gaussian, the values $\hat{x}_T$, $\hat{v}_T$ are samples from normal distributions $N(\mu = x_T, \sigma_x)$, $N(\mu = v_T, \sigma_v)$, where σ_x (here, for illustration $\sigma_x = 4cm$), σ_v (here $\sigma_v = 5cm/s$) reflect the level of noise. Further, we assume the brain encodes not only point estimates ($\hat{x}_T$, $\hat{v}_T$), but also their uncertainty—the estimates are encoded as normal distributions $N(\mu = \hat{x}_T, \sigma = \sigma_x)$ and $N(\mu = \hat{v}_T, \sigma = \sigma_v)$, respectively (see Derivation of the normative impact prediction model for details). **Displacement calculation**. According to ΔT, the object displacement distribution is $N(\mu = \Delta T \cdot \hat{v}_T, \sigma = \Delta T \cdot \sigma_v)$. **Future position estimation**. Knowing the current

position and displacement during ΔT, the position at time $T + \Delta T$ is calculated as $position_{T+\Delta T} = position_T + displacement$. Consequently, the distribution of possible future positions $\hat{X}_{T+\Delta T}$ is $N(\mu = \hat{x}_T + \Delta T \cdot \hat{v}_T, \sigma = \sqrt{(\Delta T \cdot \sigma_v)^2 + \sigma_x^2})$. **Hit probability estimation**. As the body position is at $x = 0$, the object will hit the body if its position is equal or smaller than zero (see the green part of the distribution). Therefore, the estimated probability of body hit (i.e., $y = 1$) is $P(y = 1|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v)) = P(\hat{X}_{T+\Delta T} \leq 0)$. The probability estimation of no contact is $P(y = 0|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v)) = 1 - P(y = 1|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v))$, which corresponds to the crimson part of the distribution. **Bayesian decision/prediction**. Following Eq (1), a prediction y^*_{pred}—which minimizes the expected loss—is calculated. See S1 and S2 Files for details of the computation.

https://doi.org/10.1371/journal.pcbi.1010464.g001

function—loss is 0 if the prediction y_{pred} equals y, 1 otherwise—the optimal prediction (i.e., minimizing expected loss) is to predict the state with the highest probability. More generally, however, a number of different loss functions could be used. Here, we define a fairly general loss function as,

$$loss(y, y_{pred}) = FP \max(0, y_{pred} - y)^2 + FN \max(0, y - y_{pred})^2, \quad (3)$$

where FP, $FN \in [0, \infty]$ are respectively the false positive and false negative factors, and max(0, x) is a function which outputs x for $x \geq 0$ and 0 for $x < 0$. In other words, FP determines the penalty, or cost, associated with predicting impact when none occurs, and FN determines the penalty associated with not predicting impact when one does occur.

Throughout the article, we typically assume $FN > FP$, as we focus on defensive PPS and given that it is arguably better to erroneously predict tactile activation (FP) than it is to experience impact on our bodies without predicting it (FN) (see *The Precautionary Principle*). In this case an impact prediction minimizing the expected loss is performed. We typically use $FN = 5$; $FP = 1$. This choice is arbitrary and was chosen experimentally. The effect of different choices (1, 5, 100) is illustrated in Section A graded PPS "boundary"—Effect of sensory uncertainty and cost of false negative prediction. We did not study the case where $FN < FP$, which may correspond to appetitive actions like reaching or grasping (see also [34]), but such values can be readily tested with the current model. Furthermore, for the special case when $FP = FN$, the model performs *optimal impact prediction*—the error between the prediction and the actual state is minimized. In this case, the optimal prediction is equal to the hit probability estimation. In what follows, we complement every graph in the main body of the article (with $FN = 5$; $FP = 1$) with a twin figure in the S1–S5 Figs where $FN = FP = 1$.

Putting the above together (estimated probability of touch and loss function), we may write the full expression (see Eq (6) for the derivation),

$$\begin{aligned} L((\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v), y_{pred}) &= P(y = 1|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v)) \cdot loss(y = 1, y_{pred}) + \\ P(y = 0|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v)) &\cdot loss(y = 0, y_{pred}) = \\ P(y = 1|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v))FN(1 - y_{pred})^2 &+ (1 - P(y = 1|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v))FPy_{pred}^2 \end{aligned} \quad (4)$$

In what follows, we perform simulations to compare properties of this normative model of impact prediction with known properties of PPS encoding.

A graded PPS "boundary"—Effect of sensory uncertainty and cost of false negative prediction

The study of PPS was jump-started by the realization that the primate brain has a set of neurons encoding multisensory objects when these are near from the body [2, 6, 10, 30, 35, 36]. Thus, first and foremost, if the impact prediction model accounts for PPS, it ought to differentiate between near and far spaces. In addition, more recently authors have highlighted that this

Table 1. Baseline model parameters. Negative values for velocity v_T indicate objects approaching the body, while positive values would indicate objects receding from the body. In simulations we manipulate each of these parameters, except for σ_x and *FP*.

velocity	$v_T = -25cm/s$
velocity estimation uncertainty	$\sigma_v = 20cm/s$
position estimation uncertainty	$\sigma_x = 2.5cm$
false negative factor	$FN = 5$
false positive factor	$FP = 1$
prediction time step	$\Delta T = 0.5s$

https://doi.org/10.1371/journal.pcbi.1010464.t001

PPS "boundary" is not all-or-none, but graded [37]. Thus, in a second step we question if and how the impact prediction model allows for graded PPS "boundaries".

First, we build a baseline model with the parameter values listed in Table 1.

As shown in Fig 2, the model generates predictions of contact y^*_{pred} that grow gradually with object proximity to the body. Further, it differentiates between a "far space" where touch is not likely to occur, and a "near space" where touch is highly likely to occur. If we consider the PPS "boundary" as the first value of predicted impact where $mean(y^*_{pred}) > 0.01$ (see [14], Fig 17 & 18 for a similar approach). With this basal configuration the impact prediction model specifies a "boundary" between far and near space at about 50cm from the body.

An alternative operationalization of the PPS "boundary" used in the literature is the midpoint of a sigmoid function (e.g., [29, 33, 38]). Interestingly, close examination not solely of the mean response (solid line), but also of the variability (blue dots) with the model (Fig 2) seems to indicate that impact prediction estimates are most variable near the PPS "boundary" region. We examined if this property was apparent in empirical data by re-analyzing data from [39]. In this study, human observers (n = 19) were asked to respond to touch as quickly as possible as task irrelevant visual stimuli approached their body in virtual reality. In Fig 3A we show that reaction times to visuo-tactile stimuli were faster than to tactile stimuli alone. Further, this multisensory facilitation was most apparent as visual stimuli were near the body—

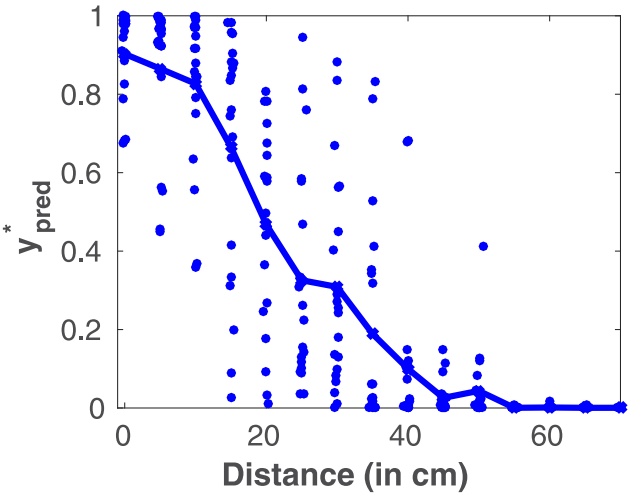

Fig 2. PPS as optimal impact utility prediction for baseline parameters. Blue dots—20 for each distance—are individual predictions (samples) of y^*_{pred}. Blue line—mean of 20 repetitions. Parameters used are in Table 1. See S1 Fig for a version with $FN = FP = 1$.

https://doi.org/10.1371/journal.pcbi.1010464.g002

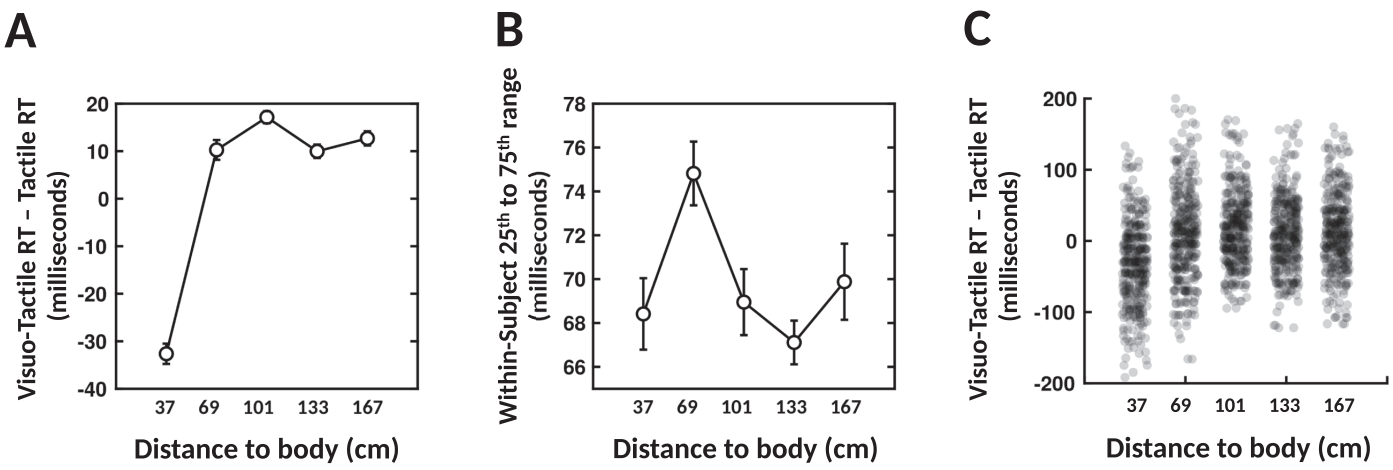

Fig 3. Variability in multisensory facilitation as a function of distance from the self–empirical data. New evaluation of data from Masson et al. [39]. (A) Visuo-tactile facilitation of reaction times (RT) as a function of distance to the body—means and standard errors across subjects. (B) Within-subject variability of reaction times. (C) Aggregate subject, combining visuo-tactile RT facilitation across all subjects.

https://doi.org/10.1371/journal.pcbi.1010464.g003

indexing the encoding of PPS. In this dataset, the PPS "boundary" was located between the first and second visuo-tactile distance indexed. Most importantly, in Fig 3B we quantified variability in reaction times, at a single subject level. That is, while reports (e.g., [15, 16, 40, 41]) typically illustrate between-subject variability (for instance by showing standard errors of the mean across subjects), there is no quantification of within-subject variability. Here, for each subject we measure the range between the 25th and 75th percentile of their reaction times, for a given subject and distance. Fig 3B depicts the mean of these ranges across subjects, and shows that within-subject variability peaked at the second distance indexed. In Fig 3C we show all reaction times measured, again showing the largest range at the second distance index. Altogether, the empirical results concur with the modeling prediction that within-subject variability is largest near the PPS "boundary".

Next, we questioned if and how this model may account for steepness in the PPS boundary, as well as for changes in its size—the most common experimental finding (e.g., PPS expanding with tool use [42], or during walking [40], or bodily illusions [41]). Conveniently, this normative model of impact prediction in essence has two degrees of freedom: (1) the uncertainty associated with perceptual observations, and (2) the ratio of *FP, FN*, dictating an appraisal of the danger associated with the objects approaching the body. For simplicity, we refer to these degrees of freedom respectively as a 'sensory' and 'cognitive' node, yet it is well established that socio-emotional contexts and motor constraints/possibilities impact our appraisal of the value of objects in our environment (e.g., see [4, 5, 37]). One additional parameter is the ΔT. This is the prediction time step of the model—a time interval for which contact estimation is performed. The object may hit the body at any moment within this interval. Its effects will be studied in Section PPS shape modulated by prediction time step. The rest of parameters (e.g., x_T, v_T) depend on the physical state of the world.

In turn, in Fig 4A and 4B we respectively manipulate σ_v (5, 20, and 35 cm/s) and *FN* (1, 5, and 100). As shown in Fig 4A, changes in sensory uncertainty lead to concurrent increase in PPS size (i.e., the first distance at which y^*_{pred} is higher than 0.01 being farther and farther in space), and a decrease in the sharpness of its boundary. On the other hand, increasing *FN* (while maintaining *FP* constant at 1), Fig 4B, increases the size of PPS while leaving the shape of its boundary virtually unchanged. Together, these results demonstrate that the normative

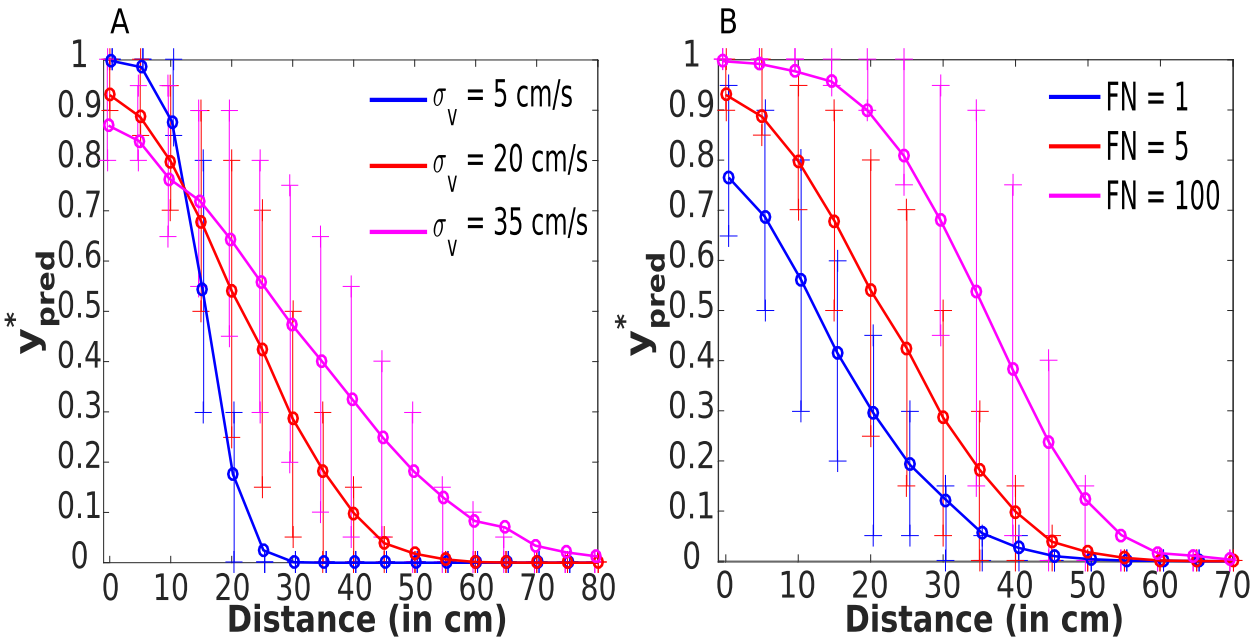

Fig 4. Effect of stimulus uncertainty and the False Negative (FN) penalty parameters. Dependency between the mean of 1000 predicted tactile activations y^*_{pred} (for each distance) and distance x_T (in centimeters) of the stimuli from the body. The symbols "+" indicate 25th and 75th percentiles which are calculated from 1000 predicted values y^*_{pred} for each distance. **(A) The size of PPS and slope of its boundary are modulated by σ_v. (B) The size of PPS, but only minimally the slope of its boundary, are modulated by *FN*.** Parameters used are in Table 1 (except for σ_v in (A) and *FN* in (B)). See S2 Fig—the right upper panel—for a version of subfigure **A** with $FN = FP = 1$.

https://doi.org/10.1371/journal.pcbi.1010464.g004

model of impact prediction not only differentiates between a near and far space but also shows that both sensory and higher-level value attributes [37] may impact the size and shape of PPS. In S6 Fig we explore how σ_v, ΔT and *FN* may simultaneously impact the gradient of the PPS boundary and PPS size.

Finally, note that the observed effect that increasing perceptual uncertainty increases the PPS size is apparent when the PPS boundary is operationalized as the farthest distance for which $mean(y^*_{pred}) > 0.01$. If instead the midpoint of a sigmoid function is estimated and used as a proxy for PPS size, the effect is significantly smaller. For the special case where $FP = FN = 1$, S2 Fig, top panels, there is no effect on "PPS size" at all.

PPS encoding and object velocity

In addition to defining a graded separation between near and far spaces, PPS encoding is also modulated by the characteristics of nearby external objects, such as their velocity [10, 14], movement direction [6, 29, 30], and valence [31, 32]. In the next three sections we tackle each of these properties in turn.

PPS size expands with the increasing velocity of incoming stimuli [10, 14]. Hence, we questioned whether our model recapitulates this finding. The simulation setup mimicked the setting from [14], with an object approaching the observer with a fixed velocity v_T equal to -25 or -75 cm/s (looming toward the subject). As shown in Fig 5, the impact prediction model inherently shows the dependency between distance of the object to the observer and impact prediction y^*_{pred} for both velocities. In fact, if we again operationalize the PPS "boundary" as the farthest distance for which $mean(y^*_{pred}) > 0.01$, our simulation roughly corresponds to the size of PPS empirically measured around the face (i.e., 52 cm for 25 cm/s and 77 cm for velocity 75

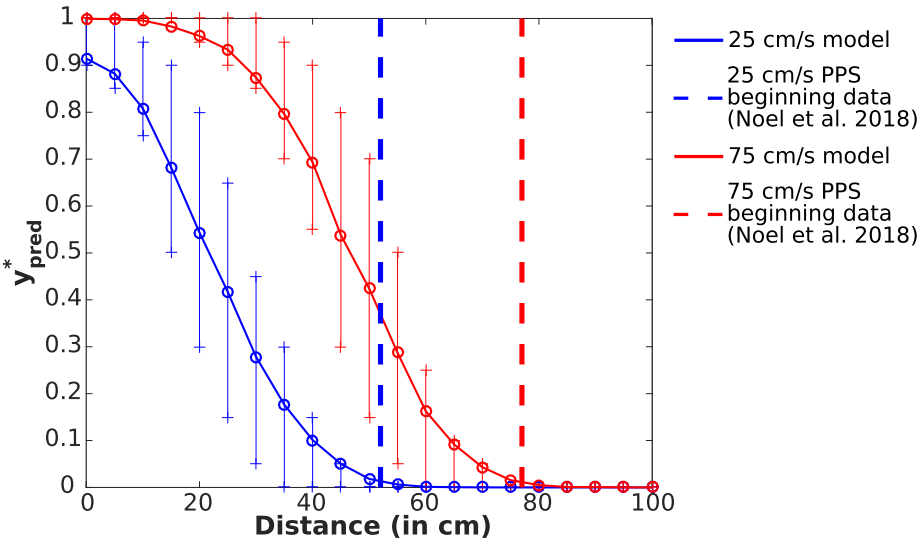

Fig 5. Comparison of PPS sizes for object velocities of -25 and -75 cm/s. Dependency between the mean of 1000 repetitions of impact predictions y^*_{pred} and distance x_T (in centimeters) between the stimuli and body, for different object velocities. The symbol "+" indicates 25th and 75th percentiles which are calculated from 1000 predicted values y^*_{pred} for each distance. Notice that the beginning of PPS—defined as the farthest distance for which $mean(y^*_{pred}) > 0.01$—roughly corresponds to the PPS beginning around the face determined by [14]. Except for the velocity $v_T = -75cm/s$, the baseline parameters from Table 1 are used. See S3 Fig for a version with $FN = FP = 1$.

https://doi.org/10.1371/journal.pcbi.1010464.g005

cm/s; [14]). Thus, while Noel et al. [14] hypothesize that the enlargement of PPS during increasing object velocity is due to neural adaptation (i.e., progressively stronger inputs are needed to drive a neuron that has been active for a given time), here we are agnostic about the neural implementation and instead show that the physics of our environment naturally leads to an enlargement of PPS with increased object velocities under a framework of impact prediction (see [17] for a similar demonstration that PPS encoding results from the physics of the environment wherein touch is more likely to occur when objects are near the body).

PPS encoding and looming versus receding objects

PPS encoding is also modulated by the movement direction of objects in the external environment. Namely, neurons mapping PPS are most readily driven by looming, as opposed to receding sensory stimuli [6, 30]. Here we replicate this situation by simulating objects moving with negative (toward the body) or positive (away from the body) velocities. Further, to extend on the empirical data and generate predictions for further experiments, we also simulate objects moving at different speeds (v_T = 12.5cm/s or 25cm/s) and with different levels of sensory uncertainty (σ_v = 5cm/s, 20cm/s, or 35cm/s), both while approaching or receding from the observer.

As expected, the results demonstrate that when objects loomed toward the body, the predicted tactile activation was higher than when it receded from the body—see Fig 6 and compare the curves corresponding to the same speed v_T and uncertainty σ_v but with opposite directions. Most importantly, our model still generated non-zero y^*_{pred} when the object recedes from the body. This is due to object position and velocity estimations having non-zero uncertainties σ_x, σ_v. Namely, predicted contact for a receding stimulus would be zero if the location and velocity of stimuli were known without any uncertainty (i.e., σ_x and σ_v were zeros). The fact that the current simulations and Bayesian Decision Theory are able to recapitulate not

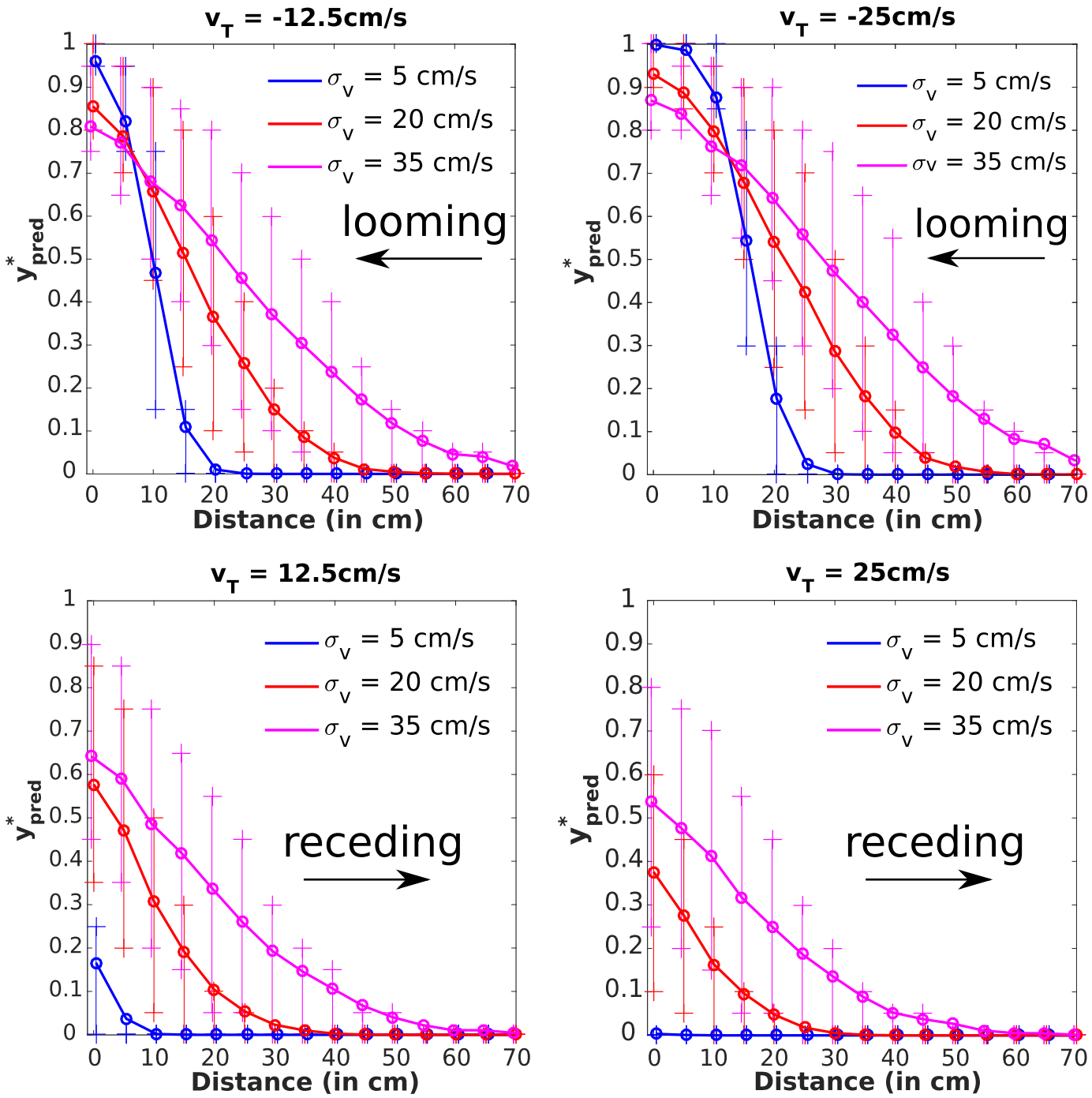

Fig 6. A looming stimulus leads to a higher response than a receding one. The stimulus is looming (receding) to (from) the body with velocity v_T size 12.5 or 25 cm/s. The horizontal axis is the distance x_T of the stimulus from the body. The vertical axis corresponds to the impact prediction y^*_{pred}—for the mean and 25th/75th percentiles of 1000 predictions for each distance. (**Left column**) The speed of the stimulus was $v_T = \pm 12.5 cm/s$. Although the prediction values were significantly smaller for the receding movement, it was still slow enough to get significant impact prediction values even for the receding movement. With increasing velocity uncertainty σ_v of the stimulus, the prediction values increased. (**Right column**) Speed was increased to $v_T = \pm 25 cm/s$. This led to reduction of impact prediction values y^*_{pred} for the receding movement compared with the smaller speed case. The parameters not listed here take values from Table 1. See S2 Fig for a version with $FN = FP = 1$.

https://doi.org/10.1371/journal.pcbi.1010464.g006

only a response to looming, but also to receding stimuli, supports the hypothesis that PPS reflects a stochastic computation of impact prediction.

Further, we can use this framework to make specific predictions for future empirical work. Namely, according to this model, when looming stimuli increase in speed, PPS expands (see above). However, when receding stimuli increase in speed, there is a negligible probability that at the next time-point the object will make contact with the body (i.e., increased velocity away from the observer offsets the effect of object position being uncertain). Thus, while PPS should expand with increasing velocity of looming stimuli [6, 29, 30], there should be no discernible PPS gradient with fast receding stimuli. Similarly, the ability to delineate a PPS boundary should decrease with increasing sensory uncertainty during looming object trajectories (i.e., the boundary becomes shallower). To the best of our knowledge, these experimental conditions (looming and receding object trajectories during different velocities and uncertainty) have not been tested, and will constitute an important future test in ratifying PPS as predicting future impact.

PPS encoding and object value

The approach of dangerous objects leads to an expansion of PPS (see e.g., [31, 32, 38, 43]). Within our normative impact prediction model, this effect would *a priori* seem most naturally accommodated by a change in *FN*. However, it may also be argued that greater encoding resources may be attributed to the encoding of dangerous objects, for instance via attentional mechanisms (see [44]), and hence reduce σ_v.

As demonstrated above (Fig 4), these competing hypotheses conveniently lead to different predictions. If the expansion of PPS during approach of dangerous objects is due to an increase in *FN* (Fig 4B), we should observe a change in PPS size, with nearly no corresponding change in its gradient. On the other hand, if σ_v decreases (Fig 4A), the PPS "boundary" becomes sharper, and importantly, this leads to shrinking rather than expansion of the size of PPS.

Taffou and Viaud-Delmon [43] used ecological auditory stimuli (dog growling vs. sheep bleating) and reported that PPS expanded in the dog condition, specifically in subjects scared of dogs. They did not explicitly report on the gradient of PPS, yet visual examination suggests no difference between dog and sheep conditions. This—PPS expansion and no apparent change in gradient—putatively suggests that the effect reported in [43] is "cognitive" in nature (i.e., originates from the loss function, *FN*) Importantly, this effect, as interpreted under the current modeling framework also highlights a critical element of the Bayesian observer performing contact prediction; namely that beyond optimizing the prediction of the probability that touch will occur, PPS encoding also ought to optimize the utility associated with impact prediction.

Ferri et al. [38] ratify the conclusion from [43], while also directly comparing ecological and artificial stimuli. In a first experiment, the authors present artificial sounds associated with negative and neutral valence—broadband Brown and White noise, respectively (see [38]). The results show both an expansion and sharpening of PPS during the negative-valence condition. Our model would predict that this may be a simultaneous "sensory" effect driving the change in PPS boundary steepness and a "cognitive" effect driving the PPS expansion and overriding any shrinking due to the new shape of the PPS boundary as a result of decrease in σ_v.

Together, this pattern of results highlights the importance in fully characterizing changes in PPS encoding (only when size and gradient are quantified, one can attribute these effects to "sensory" vs. "cognitive" in nature). Further, they suggest that when using ecologically valid sounds—but not artificial stimuli—, enlargements of PPS are most likely due to modulations in the loss function and not low level sensory components. Lastly, these results highlight that,

according to the current framework, not all previously reported characteristics of PPS encoding may be explained by either environmental factors or changes in the probability of touch occurring. Instead, impact prediction must also account for the value attributed to environmental objects [37].

PPS size across different body parts

Beyond defining a graded boundary between near and far space that is modulated by context, another important characteristic of PPS is that it is dependent on body-part, with PPS growing in size from hand to face to torso [33]. The differing size of PPS across body parts is unlikely due to modulations in the sensory uncertainty associated with object position or velocity (σ_x and σ_v) given that approaching objects are perceived by exteroception (i.e., vision or audition) which is common across body parts. In theory, the ratio between *FN* and *FP* could account for the different sizes of PPS across body parts, but we would have to posit *FN* being larger for the torso than the face, and it is not immediately clear why this would be the case. Perhaps the most parsimonious explanation would be that the difference in PPS size simply reflects differences in body-part size. In order to test this possibility, we extend the model from 1-dimensional to 3-dimensional. We only model the face and torso in this section.

To extend the model to three dimensions, we generalized 1D position and velocity to 3D vectors and the border of a body part is generalized to a 2D rectangle enclosed in 3D space—only the "collision plane", not the depth of the body part is considered; see Fig 7. The details are in Section Extension to 3D space. We approximated the face by a rectangle with size [25*cm*, 25*cm*], and the torso by a rectangle with size [50*cm*, 50*cm*]. In contrast to the 1D scenario, now the object *can miss* the body part, which decreases the probability of hit. In all

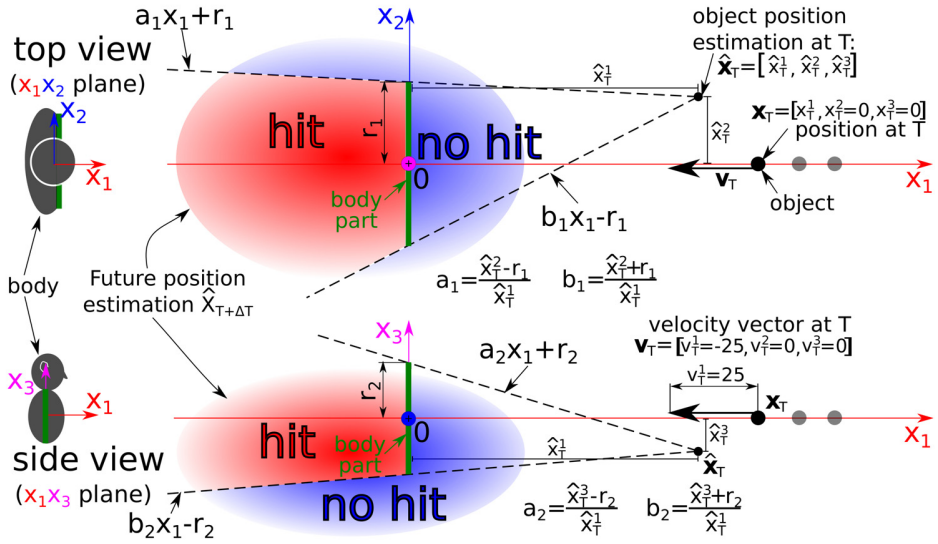

Fig 7. 3D experimental scenario. An object is looming to a body part (2D rectangle with size $[2 \cdot r_1, 2 \cdot r_2]$ enclosed in 3D space). As the object moves along the x_1 axis, it has position $\mathbf{x}_T = [x_T^1, x_T^2 = 0cm, x_T^3 = 0cm]$ and velocity $\mathbf{v}_T = [v_T^1 = -25cm/s, v_T^2 = 0cm/s, v_T^3 = 0cm/s]$ at time T. As the uncertainty in position estimation is nonzero ($\boldsymbol{\sigma}_x = [\sigma_x^1 > 0, \sigma_x^2 > 0, \sigma_x^3 > 0]$), the point position estimation $\hat{\mathbf{x}}_T = [\hat{x}_T^1, \hat{x}_T^2, \hat{x}_T^3]$ does not correspond to $\mathbf{x}_T$. Future position estimation $\hat{X}_{T+\Delta T}$ with a multivariate normal distribution is then calculated (see Section Extension to 3D space for details). The red area of $\hat{X}_{T+\Delta T}$ corresponds to the probability estimation of hit—the body part is on the path between $\hat{\mathbf{x}}_T$ and each point of the red area. On the contrary, the blue area corresponds to no hit of the body. (Top) Top view. (Bottom) Side view. The silhouette's reference frame (left) is placed to the torso.

https://doi.org/10.1371/journal.pcbi.1010464.g007

experiments, the object is moving along x_1 axis to the center of the body part (see Fig 7). Therefore, if the position and velocity uncertainty in the vertical and horizontal axis are zero ($\sigma^{2,3}_{x,v} = 0$), the probability estimation of hit is the same as in the 1D case, because missing the body part on the left/right or over/under it is excluded. This means that the variables related to the first dimension (e.g., x^1_T, v^1_T, σ^1_x) are equivalent to the variables of the 1D model (e.g., x_T, v_T, σ_x). On the other hand, if the horizontal ($\sigma^2_{x,v}$) or vertical ($\sigma^3_{x,v}$) uncertainty increases, there is a corresponding stochastic estimate that the object may miss the body part and hence the estimation of probability of hit and of y^*_{pred} goes down.

Experiments with this model are shown in Fig 8. In the first experiment, we used baseline parameters from the 1D case (see Table 1) and manipulated horizontal (axis x_2) and vertical (axis x_3) position and velocity estimation uncertainties (first row—$\sigma^1_v = 20cm/s$—in Fig 8). For some settings of perceptual uncertainty, there is a difference in PPS size between the face and torso. However, for the torso, the beginning of PPS is still much smaller compared to the empirical value (72cm from [33]). In an effort to come close to the empirical values, we increased the velocity uncertainty in the first dimension from the baseline value to $\sigma^1_v = 30cm/s$, leading to a general expansion of PPS (similarly to the experiment from Fig 6). For position and velocity uncertainties in the other dimensions, $\sigma^{2,3}_x = 5cm, \sigma^{2,3}_v = 40cm/s$ (purple curve in Fig 8), the beginning of face and torso PPS roughly fit empirical estimations (torso 72cm [33], face 52cm [14]). Thus, to fit empirical data, large horizontal and vertical velocity uncertainty $\sigma^{2,3}_v$ and small horizontal and vertical position uncertainty $\sigma^{2,3}_x$ are necessary. If the horizontal and vertical position uncertainty is further increased to $\sigma^{2,3}_x = 10cm$, the maximal value of y^*_{pred} is only 0.6 even for zero distance from the face, which would predict bigger reaction times in close proximity for the face than for the torso. We speculate that this is not plausible.

Two additional observations are in order. First, interestingly, our results suggest that horizontal and vertical uncertainty matters more for small body parts—something that can be empirically tested. Second, for low values of horizontal and vertical uncertainty, the 3D model for the torso has very similar PPS size and shape as the 1D case. Thus, a 3D model may often not be necessary.

PPS shape modulated by prediction time step

An alternative parameter that could potentially influence the different extent of PPS is the prediction time step parameter ΔT (in our model it was fixed to 0.5s). It may be interpreted as the time the agent needs to perform a defensive action that will protect the body part threatened by the impending collision. The effects of $\Delta T \in \{0.25, 0.5, 1\}s$ on the 1D model are shown in S7 Fig (for the corresponding figure with $FN = FP = 1$ see S8 Fig). Depending on the body part and the action, the "time constant" may differ. For example, blinking to protect the eyes will be faster than squatting to protect the whole torso. To explore this hypothesis, we performed an experiment with $\Delta T = 0.5s$ for the face and $\Delta T = 0.75s$ for the torso on the 3D model—see Fig 9. It is apparent that the ΔT parameter is very effective in shifting the PPS boundary.

Discussion

Understanding how observers avoid collision with approaching environmental objects potentially harming their bodies is of paramount importance in furthering our understanding of self-environment interactions. It has long been postulated that neurons encoding for our PPS may play a critical role in this computation [4, 9, 14, 45, 46]. Yet, there has been no formal, normative demonstration. In turn, the major contribution of the current work is the

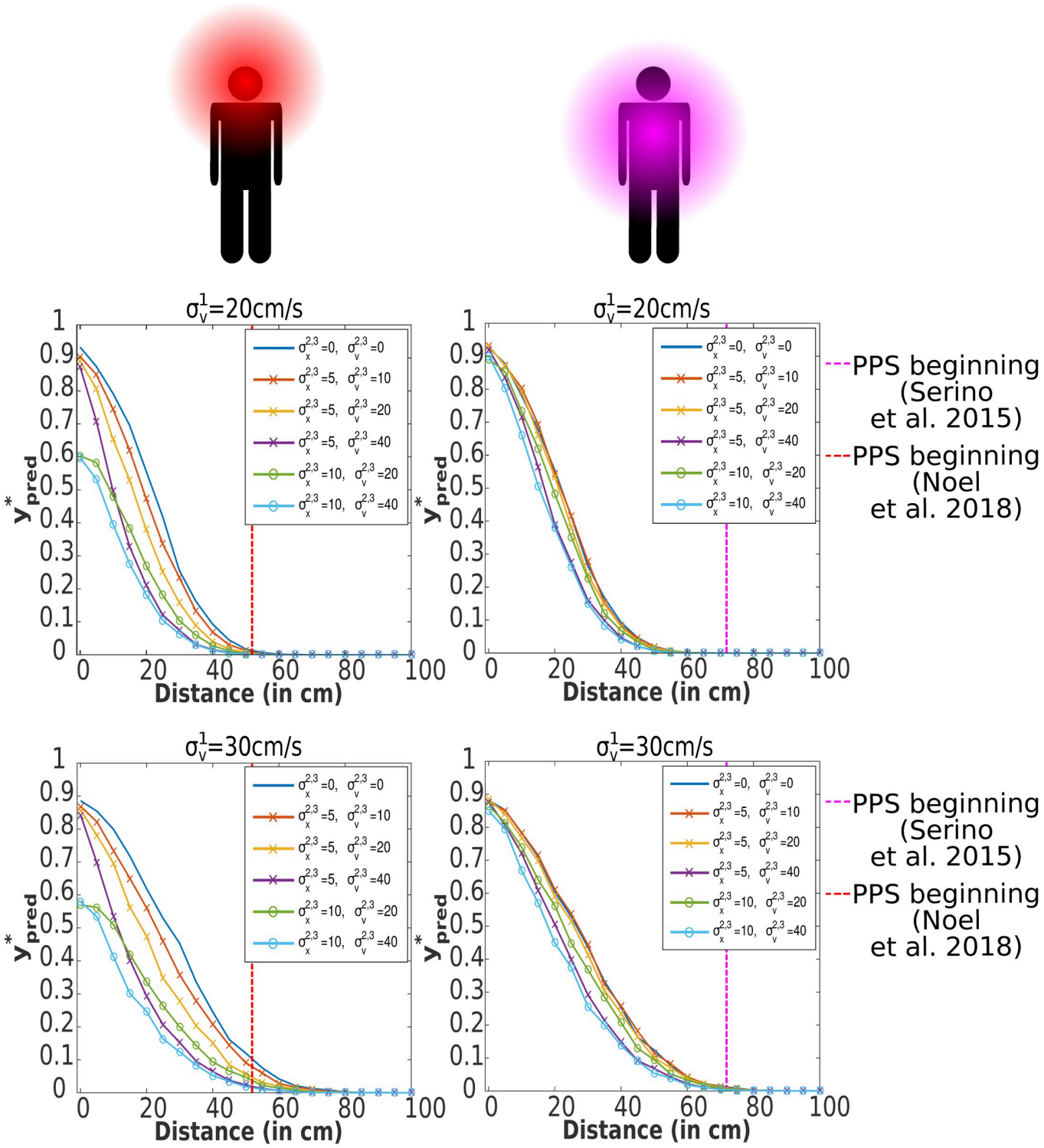

Fig 8. Modulation of PPS size by body part size in a 3D model (face and torso). For this experiment, 3D model was used (see Extension to 3D space). Dependency between distance of stimuli from body and the mean of 1000 impact predictions y^*_{pred} for each distance and for PPS representation around the face (body part size [25cm, 25cm]) and trunk (body part size [50cm, 50cm]). The object is moving along the x_1 axis ($\mathbf{x}_T = [x^1_T, x^2_T = 0, x^3_T = 0], \mathbf{v}_T = [v^1_T = -25cm/s, v^2_T = 0, v^3_T = 0]$). Position and velocity estimation uncertainty for the first dimension are $\sigma^1_x = 2.5cm$ and $\sigma^1_v = 20cm/s$ for the first row, $\sigma^1_v = 30cm/s$ for the second row. The uncertainties in the other two dimensions $\sigma^{2,3}_{x,v}$ (in cm or cm/s) are varied through the experiments. All other parameters are the baseline parameters from Table 1. The vertical dashed lines correspond to the estimations of the beginning of PPS from [14, 33]. See S4 Fig for a version with $FN = FP = 1$.

https://doi.org/10.1371/journal.pcbi.1010464.g008

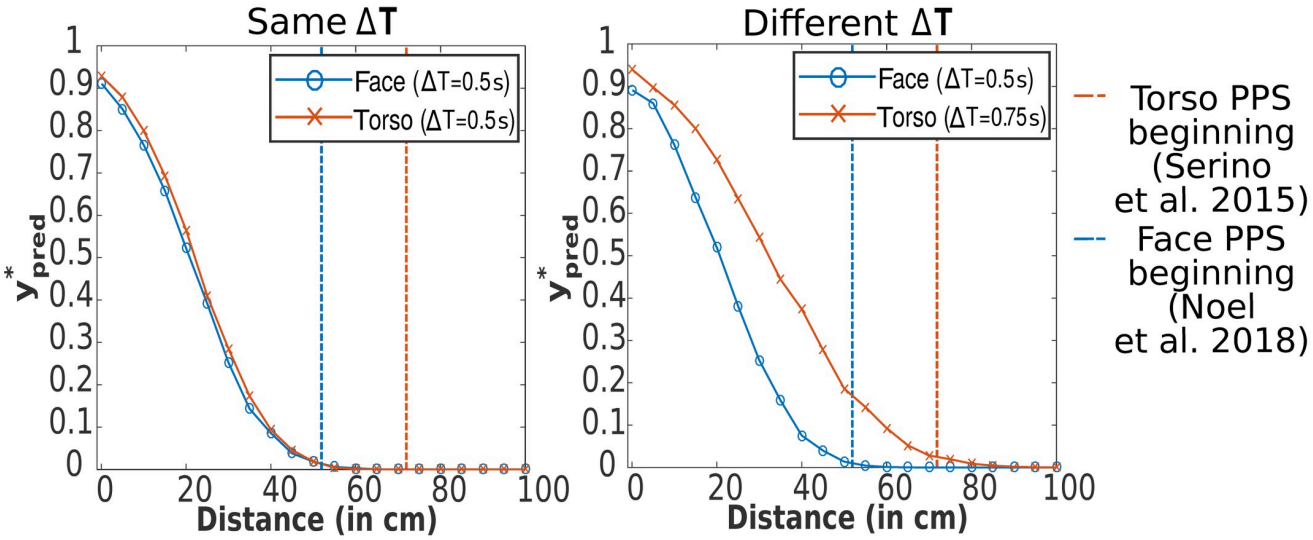

Fig 9. Modulation of PPS for face and torso in 3D model by prediction time step. Dependency between distance of stimuli from body and the mean of 1000 impact predictions y^*_{pred} calculated by the 3D model (see Section Extension to 3D space) for each distance and for PPS representation around the face (body part size [25cm, 25cm]) and trunk (body part size [50cm, 50cm]). Baseline parameters (see Table 1) were used. Horizontal and vertical uncertainties were set to $\sigma_x^{2,3} = 5cm, \sigma_v^{2,3} = 5cm/s$. For a detailed experiment description see Fig 8. **(Left)** Prediction time step ΔT is same for both body parts (baseline value). The vertical and horizontal uncertainties are not large enough to cause different sizes for both body parts. **(Right)** The prediction time step is higher for the torso ($\Delta T = 0.75s$) than for the face. In this setting, the PPS beginnings of both body parts fit roughly the empirical estimations. The vertical dashed lines correspond to the PPS beginning estimations from [14, 33]. See S5 Fig for a version with $FN = FP = 1$.

https://doi.org/10.1371/journal.pcbi.1010464.g009

derivation of a Bayes optimal model of impact prediction that consists of impact probability estimation and a cost function simulating the utility/penalty for the agent incurred by the impending collision. Supporting the hypothesis that PPS encodes the prediction of future contact, in a value-dependent manner, the normative model of impact prediction can recapitulate several of the defining characteristics of PPS: (i) a graded delineation of near and far space [37], a preference for (ii) approaching [6, 29, 30] and (iii) rapidly moving [10, 14] stimuli, (v) a scaling of the "boundary" differentiating near and far space as a function of the valence attributed to the approaching object [31, 32], and finally (v) differing sizes for different body parts [33]. The model also makes a set of concrete and testable hypotheses for future work. For instance, the fact that stimuli velocity ought to impact PPS delineation differently for looming and receding trajectories (see Fig 6), the fact that perceptual uncertainty ought to have an impact on PPS size and boundary shape (see Fig 4A) and that perceptual uncertainty in orthogonal directions to the looming object impacts more the characteristics of PPS for smaller rather than larger body parts (Fig 8), and finally, the fact that "sensory" and "cognitive" effects ought to shape PPS encoding differently (compare Fig 4A and 4B).

Interestingly, the derivation highlights two major factors (beyond the environmental, such as the position and velocity of incoming stimuli, as well as the size of body parts) that may largely determine the shape and size of PPS. First, aspects related to the loss function—the value attributed to false positive vs. false negative detection of contact (see [37] for an opinion piece proposing a value-based theory of PPS). This loss function is likely modulated by social, emotional, motor, attentional, and even reflex-like computations that ascribe a value to, or a danger associated with, objects and events in the environment (see [4, 5] for further discussion). Second, aspects related to the precision with which an observer may estimate the position and velocity of the approaching object and self-position. Conveniently, these two factors affect the overall size of PPS (e.g., the central point of a sigmoidal function differentiating

between the near and far space) and its gradient (e.g., the slope of the sigmoid) differently. While the value-based computation may modulate the overall size of PPS, it only minimally affects the gradient between near and far space. On the other hand, if an enlargement of PPS is due to changes in low-level sensory uncertainty, by necessity this has to be accompanied by a flattening of the curve differentiating between the near and far space. The differing effect engendered by changes in the loss function vs. computing the probability of contact should allow researchers to attribute their empirical effects to one or the other component of the normative impact prediction model. In S6 Fig, we provide 3D plots illustrating the effects of velocity uncertainty (σ_v), false positive cost (FN), and prediction time step (ΔT) on the slope of the PPS boundary and its size.

Manipulations intended to affect the loss function are commonplace in PPS research [31, 32]—even if not necessarily conceived as such. For instance, researchers have presented observers with sights or sounds of objects approaching with either a positive, neutral, or negative valence. Examining this literature under the current framework suggests that while ecological stimuli may in fact affect solely the loss function (i.e., changes in the false negative parameter, modulating only PPS size but not the shape of the boundary), artificial stimuli may affect both value-based computation, as well as the precision of sensory representations (see PPS encoding and object value).

More notoriously, the current framework points to a large empirical void. That is, while a critical element of the current model, there is a lack of studies examining how sensory uncertainty—by e.g., varying size, contrast, adding observation noise, or making the approach trajectory variable—may affect PPS (but see Huijsmans et al. [7] for a recent exception). The normative model of impact prediction would hypothesize that more uncertain stimuli should lead to a larger PPS, depending on how the size of PPS is operationalized—cf. Section A graded PPS "boundary"—Effect of sensory uncertainty and cost of false negative prediction. To the best of our knowledge, this has not been explicitly tested. However, Schlack et al. [47], did record from single cells in the ventral intra-parietal area—an area known to house PPS neurons (see e.g., [6])—while presenting auditory or visual stimuli (the former being more imprecisely localized in space, [48]). The authors reported larger auditory than visual receptive fields in this area, suggesting that audio-tactile PPS may be wider than visuo-tactile PPS, as the normative model of impact prediction would conjecture.

On the modeling front, PPS is commonly associated with not only defensive [6], but also with approaching behaviors [34]. Thus, in the future we may develop a full choice model, where an agent does not only predict if impact will occur or not, but could also take either avoiding or approaching actions. In this line, Roncone et al. [19] made a robot move toward or away from objects by connecting artificial "PPS neurons" to a controller. In our case, now equipped with a normative model of impact prediction, we could trigger actions based on a specific value of y^*_{pred}. Two aspects of the current work are worth highlighting in this action-oriented setting. First, here we either used a loss function where $FN > FP$ or an unbiased one ($FN = FP$). However, this need not always be the case. In particular when approaching objects, the cost associated with "miss" may be higher than that associated with a "false positive". Namely, a striking difference between "PPS for defensive behavior" and "PPS for action" may be that in the former $FN > FP$ while in the latter $FN < FP$. Second, we ought to highlight that in order to qualitatively match empirical estimates of PPS sizes across different body parts, varying the ΔT parameter was more effective than the FN/FP ratio. For defensive PPS, this parameter may be mainly motivated by the time needed to trigger and execute a protective action. This may differ for body parts—protecting the torso by moving it requires whole-body action, while hand or head could be protected relatively more easily—or even for the same body parts depending on

context, such as the character of a potential threat. For example, protecting the eyes against flying sand by blinking is more rapid than a squatting action or moving the arms in front of the face when the threat is different. Similarly, in invasive single cell recordings a striking feature of PPS neurons is their vast heterogeneity in receptive field sizes. Our current results suggest that perhaps akin to what is observed in other spatial codes (e.g., place or grid cells) this heterogeneity bears from different intrinsic time-scales of each neuron.

It is also worth noting that the our model predicts complete curves relating impact prediction and distance of the object from the body. It generates empirical predictions about how different parameters such as perceptual uncertainty or object valence modify this curve—by offsets along the distance axis, change in its slope, or their combination. To test the model predictions in real experiments, complete distance-dependent curves are desired, as opposed to simplifications defining PPS boundaries as either the farthest distance with an effect on a measured variable or as a midpoint of a fitted sigmoidal curve. Reducing the response curve to a single distance may blur the impact of the different factors.

In conclusion, we derived a normative model of impact prediction, and demonstrated that this model accounted for a number of characteristics of PPS. Further, this exercise highlighted that beyond characteristics of the environment itself, the two main factors influencing PPS size and shape are (i) the ability to represent the external environment precisely, and (ii) the value attributed to false positive and negatives. Conveniently, these factors express differently (either affecting both size and shape of PPS, or solely size), and thus researchers ought to be able to attribute their effects to one or the other. Further, our formal approach has highlighted aspects of empirical work that are still missing, most notoriously the ability to index biases and variance in PPS on the individual subject level. We hope novel methods to index PPS are developed (e.g., estimation tasks), which will allow for further joint theory—experiment examination of impact prediction and PPS encoding.

Materials and methods

Derivation of the normative impact prediction model

In line with the probabilistic (e.g., [21]) framework to perception, we propose an estimation procedure of computing the probability of future impact on the body (see Fig 1 for a schema with an example). Following the estimation procedure, Bayesian Decision Theory (e.g., [25]) is employed for impact prediction calculation.

An external object is moving on a straight line toward or away from the body. At time T, a stimulus has position $x_T \in \mathbb{R}$ (distance from the body) and moves with velocity $v_T \in \mathbb{R}$ (negative values for a looming object). We followed [21] (among others) and supposed that sensory estimations of the position $\hat{x}_T$ and velocity $\hat{v}_T$ are corrupted by Gaussian noise with variances σ_x^2 and σ_v^2, respectively. To simulate the effect of noise, $\hat{x}_T$ and $\hat{v}_T$ were obtained as samples from normal distributions $N(\mu = x_T, \sigma = \sigma_x)$ and $N(\mu = v_T, \sigma = \sigma_v)$. If the object position sample is within the body ($\hat{x}_T < 0$), it is set to $\hat{x}_T = 0.1cm$—immediately in front of the body. Notice that the higher values (e.g., auditory localization as opposed to visual localization) of the standard deviations σ_x, σ_v are related to less precise estimations.

The brain does not only encode point estimates, but also their uncertainties [21, 23, 24, 49]. Hence, we did not use only the point estimates $\hat{x}_T, \hat{v}_T$ of the position and velocity, but also included the uncertainty caused by the observation noise—the estimates of the position and velocity are encoded as normal distributions $N(\mu = \hat{x}_T, \sigma = \sigma_x)$, $N(\mu = \hat{v}_T, \sigma = \sigma_v)$, respectively.

Next, we compute an estimate of object displacement during ΔT. The displacement is encoded as $N(\mu = \Delta T \cdot \hat{v}_T, \sigma = \Delta T \cdot \sigma_v)$. Note that this estimation, based on the equation

displacement = $\Delta T \cdot$ *velocity*, is precise only if the velocity does not change during ΔT (as assumed in the current simulations and in all empirical studies of PPS with approaching objects).

Given the estimate of the initial position and displacement of the object, we can estimate its future position, $\hat{X}_{T+\Delta T}$. This position is calculated as $position_{T+\Delta T} = position_T + displacement$. In case of Gaussian random variables, this means $\hat{X}_{T+\Delta T} \sim N(\mu = \hat{x}_T + \Delta T \cdot \hat{v}_T, \sigma = \sqrt{\sigma_x^2 + (\Delta T \cdot \sigma_v)^2})$. Notice that the calculation of the overall estimation uncertainty $\sigma = \sqrt{\sigma_x^2 + (\Delta T \cdot \sigma_v)^2}$ shows that manipulations of σ_v (used in some simulations) is interchangeable with manipulations of σ_x (only ΔT has to be taken into account). Therefore, the qualitative effects engendered by manipulating velocity uncertainty σ_v in the main text can be generalized to position uncertainty σ_x. The model restricts mean of position estimation to only the space in front of the body.

We can estimate the probability of impact, $P(Y|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v))$, where $Y \in \{0, 1\}$ represents whether the object hits the body ($y = 1$) or not ($y = 0$). As the prediction is calculated before the object hits (or not) the body, the actual future impact value y is not known during the calculation. Therefore, the calculation takes into account the estimated probability $P(y|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v))$ for both possible values of y. It is estimated as $P(y = 1|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v)) = P(\hat{X}_{T+\Delta T} \leq 0)$. That is, this is the estimation that the object will be on the surface of the body or farther in space (see Fig 1) at time $T + \Delta T$. Namely, contact of the object with the body can occur at any time between time T and $T + \Delta T$. The probability estimation that the body will not be hit is $P(y = 0|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v)) = 1 - P(y = 1|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v))$. Given the above, according to Bayesian Decision Theory [25, 26], the optimal decision—in our case the impact prediction $y^*_{pred} \in [0, 1]$—is calculated as

$$y^*_{pred} = \arg\min_{y_{pred} \in [0,1]} L((\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v), y_{pred}) \tag{5}$$

where $L((\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v), y_{pred})$ can be further expanded in the following manner by using a loss function definition

$$\begin{gathered}
L((\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v), y_{pred}) = P(y = 1|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v)) \cdot loss(y = 1, y_{pred}) + \\
P(y = 0|(\hat{x}_T, \sigma_x), \hat{V}_T) \cdot loss(y = 0, y_{pred}) = \\
P(y = 1|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v)) \cdot loss(y = 1, y_{pred}) + \\
(1 - P(y = 1|(\hat{x}_T, \sigma_x), \hat{V}_T)) \cdot loss(y = 0, y_{pred}) = \\
P(y = 1|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v))(FP \max(0, y_{pred} - 1)^2 + FN \max(0, 1 - y_{pred})^2) + \\
(1 - P(y = 1|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v)))(FP \max(0, y_{pred} - 0)^2 + FN \max(0, 0 - y_{pred})^2) = \\
P(y = 1|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v))FN(1 - y_{pred})^2 + (1 - P(y = 1|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v)))FPy^2_{pred}
\end{gathered} \tag{6}$$

A prediction, $y_{pred} = 1$ corresponds to hit prediction, is evaluated according to a function *loss*: $Y \times Y_{pred} \rightarrow [0, \infty)$ which determines the cost incurred (or penalty) when the predicted value y_{pred} does not correspond to the future tactile impact value y. In other words, the loss function reflects the difference between the predicted tactile activation and the actual future tactile activation y at time $T + \Delta T$. The loss function is expressed as

$$loss(y, y_{pred}) = FP \max(0, y_{pred} - y)^r + FN \max(0, y - y_{pred})^r \tag{7}$$

where $FP, FN \in [0, \infty]$ are respectively the false positive and false negative factors, $\max(0, x)$ is a function which outputs x for $x \geq 0$ and 0 for $x < 0$. The parameter $r \in (0, \infty)$ shapes the loss function. Throughout the simulations, we maintained it fixed to $r = 2$. If the prediction matches the actual impact value, the loss will be 0. Instead, if $y_{pred} > y$, then the loss function (7) is reduced to $loss(y, y_{pred}) = FP(y_{pred} - y)^2$ and the maximal value is reached when tactile contact is predicted ($y_{pred} = 1$) but does not happen ($y = 0$). Lastly, if $y_{pred} < y$, then the loss function (7) is equal to $loss(y, y_{pred}) = FN(y - y_{pred})^2$ and the loss is maximal when contact occurs ($y = 1$) without a prediction of this happening ($y_{pred} = 0$). We suggest that the loss during FN cases is higher than during FP cases because objects making contact with the body without any prediction—thus no defensive action—may be more harmful than making predictions of contact that do not in fact occur.

Note that the prediction is optimal in relation to the estimated probability $P(y|(\hat{x}_T, \sigma_x), (\hat{v}_T, \sigma_v))$ of (no) impact given the object position and velocity estimations. Because these sensory estimations are stochastic (point estimations $\hat{x}_T, \hat{v}_T$ of x_T, v_T are corrupted by Gaussian noise), there are multiple predictions y^*_{pred} for one position x_T and velocity v_T and all of them are optimal in relation to the object position and velocity estimations $N(\mu = \hat{x}_T, \sigma_x), N(\mu = \hat{v}_T, \sigma_v)$ of x_T and v_T, respectively.

Extension to 3D space

The model proposed above is one-dimensional. We extended this model to three dimensions. It means that both position and velocity are represented by 3-dimensional vectors $\mathbf{x}_T = [x^1_T, x^2_T, x^3_T]$ and $\mathbf{v}_T = [v^1_T, v^2_T, v^3_T]$. In our model, the movement in each dimension is treated equivalently to the movement in the 1D model and independently on other dimensions (see the selected reference frame in Fig 7). Therefore, position and velocity point estimates $\hat{\mathbf{x}}_T = [\hat{x}^1_T, \hat{x}^2_T, \hat{x}^3_T], \hat{\mathbf{v}}_T = [\hat{v}^1_T, \hat{v}^2_T, \hat{v}^3_T]$ are sampled independently in individual dimensions depending on the position and velocity uncertainties $\boldsymbol{\sigma}_x = [\sigma^1_x, \sigma^2_x, \sigma^3_x], \boldsymbol{\sigma}_v = [\sigma^1_v, \sigma^2_v, \sigma^3_v]$.

The three-dimensional generalization $\hat{\mathbf{X}}_{T+\Delta T}$ of the one-dimensional future position estimation $\hat{X}_{T+\Delta T} \sim N(\mu, \sigma)$ is distributed as a multivariate normal distribution with a diagonal covariance matrix (see Fig 7)

$$\hat{\mathbf{X}}_{T+\Delta T} \sim N\left(\begin{array}{l} \mu_1 = \hat{x}^1_T + \Delta T \cdot \hat{v}^1_T, \sigma_1 = \sqrt{(\sigma^1_x)^2 + (\Delta T \cdot \sigma^1_v)^2} \\ \mu_2 = \hat{x}^2_T + \Delta T \cdot \hat{v}^2_T, \sigma_2 = \sqrt{(\sigma^2_x)^2 + (\Delta T \cdot \sigma^2_v)^2} \\ \mu_3 = \hat{x}^3_T + \Delta T \cdot \hat{v}^3_T, \sigma_3 = \sqrt{(\sigma^3_x)^2 + (\Delta T \cdot \sigma^3_v)^2} \end{array}\right) \tag{8}$$

The body part is represented as a rectangle with size $[2 \cdot r_1, 2 \cdot r_2]$ (see Fig 7). The probability of a hit is estimated as

$$\begin{aligned} P(y = 1|(\hat{\mathbf{x}}_T, \boldsymbol{\sigma}_x), (\hat{\mathbf{v}}_T, \boldsymbol{\sigma}_v)) = \int_{-\infty}^{0} \int_{b_1 x_1 - r_1}^{a_1 x_1 + r_1} \int_{b_2 x_1 - r_2}^{a_2 x_1 + r_2} f_{\hat{\mathbf{X}}_{T+\Delta T}}(x_1, x_2, x_3)\, dx_1\, dx_2\, dx_3, = \\ \int_{-\infty}^{0} \int_{b_1 x_1 - r_1}^{a_1 x_1 + r_1} \int_{b_2 x_1 - r_2}^{a_2 x_1 + r_2} f_{N(\mu_1, \sigma_1)}(x_1) \cdot f_{N(\mu_2, \sigma_2)}(x_2) \cdot f_{N(\mu_3, \sigma_3)}(x_3)\, dx_1\, dx_2\, dx_3, \end{aligned} \tag{9}$$

where a_1, a_2, b_1, b_2 are the parameters of the integration boundaries (see Fig 7 for details) and f represents the probability density function. The probability of no hit can be calculated as $P(y = 0|(\hat{\mathbf{x}}_T, \boldsymbol{\sigma}_x), (\hat{\mathbf{v}}_T, \boldsymbol{\sigma}_v)) = 1 - P(y = 1|(\hat{\mathbf{x}}_T, \boldsymbol{\sigma}_x), (\hat{\mathbf{v}}_T, \boldsymbol{\sigma}_v))$.

In our simulations, to speed up the probability calculation determined by the integral from Eq 9 and avoid problems (for example, zero horizontal and vertical uncertainties), we used

numerical calculation. We generated 10000 samples for each future position estimation. The probability was estimated as a rate of samples within the "hit" area to all samples (see the code).

Simulation details

In the simulations, we mimicked the setup of empirical reports. An object was approaching or receding from the body with constant velocity v_T. In one experimental trial, for each distance x_T (e.g., 0, 5, 10, . . ., x_{max} cm) from the body, an impact prediction y^*_{pred} was calculated. Notice that the choice of the x_{max} (beginning of the trajectory, in case of looming stimuli) did not affect the computed values of y^*_{pred}, because the predicted values depend only on the actual position and velocity (which is constant) and not on the previous trajectory.

Because the predictions y^*_{pred} differ from trial to trial—similarly to measures in experiments with human observers—multiple trials for every experimental condition were performed. To summarize multiple predicted values y^*_{pred} for each distance x_T, means of y^*_{pred} and 25th/75th percentiles for each distance x_T were calculated. In simulations, the expected loss (Eq (6)) is calculated for $y_{pred} \in \{0, 0.05, 0.1, \ldots, 1\}$ (except the experiment in Fig 2 where the granularity is 0.001) and the one with the smallest loss is then selected as the optimal value y^*_{pred}. A detailed example of y^*_{pred} calculation with all details is in S1 and S2 Files (interactive version).

Supporting information

S1 File. A detailed example of an impact prediction calculation—Interactive version.
(PDF)

S2 File. A detailed example of an impact prediction calculation. For a more interactive version see S1 File.
(PDF)

S1 Fig. A version of Fig 2 with *FN* = *FP* = 1.
(EPS)

S2 Fig. A version of Fig 6 with *FN* = *FP* = 1.
(EPS)

S3 Fig. A version of Fig 5 with *FN* = *FP* = 1. The vertical dashed lines correspond to the PPS beginning estimations from [14].
(EPS)

S4 Fig. A version of Fig 8 with *FN* = *FP* = 1. The vertical dashed lines correspond to the PPS beginning estimations from [14, 33].
(EPS)

S5 Fig. A version of Fig 9 with *FN* = *FP* = 1. The vertical dashed lines correspond to the PPS beginning estimations from [14, 33].
(EPS)

S6 Fig. Size of PPS and slope of its boundary is modulated by FN, ΔT and σ_v. Beginning of PPS is determined as the farthest distance x_T for which the mean value of 1000 y^*_{pred} samples overcomes 0.01. For slope calculation, mean values of 1000 y^*_{pred} samples for each distance x_T were used. The slope was calculated around the central value (between min and max) of the curve. Technically, the slope was negative—the values were decreasing from left to right—in all cases. To better visualize the slope, we plotted absolute values of the slope. Except for σ_v, ΔT,

FN and $\sigma_x = 0cm$, the baseline parameters (see Table 1) were used. See the code for details.
(EPS)

S7 Fig. Effect of timestep ΔT size on PPS. Dependency between the mean of 1000 predicted tactile activations y^*_{pred} (for each distance) and distance x_T (in centimeters) of the stimuli from the body. The symbol "+" indicates 25th and 75th percentiles which are calculated from 1000 predicted values y^*_{pred} for each distance. PPS size expands with increasing size of timestep ΔT (in seconds). Sharpness of the PPS boundary is decreasing with increasing size of timestep ΔT. Except for ΔT, baseline parameters are used (Table 1).
(EPS)

S8 Fig. A version of S7 Fig with *FN* = *FP* = 1.
(EPS)

Author Contributions

Conceptualization: Zdenek Straka, Jean-Paul Noel, Matej Hoffmann.

Formal analysis: Zdenek Straka, Jean-Paul Noel.

Funding acquisition: Matej Hoffmann.

Methodology: Zdenek Straka, Jean-Paul Noel, Matej Hoffmann.

Project administration: Matej Hoffmann.

Software: Zdenek Straka.

Supervision: Matej Hoffmann.

Validation: Zdenek Straka, Matej Hoffmann.

Visualization: Zdenek Straka.

Writing – original draft: Zdenek Straka.

Writing – review & editing: Jean-Paul Noel, Matej Hoffmann.

References

1. Rizzolatti G, Scandolara C, Matelli M, Gentilucci M. Afferent properties of periarcuate neurons in macaque monkeys. II. Visual responses. Behavioural Brain Research. 1981; 2(2):147–163. https://doi.org/10.1016/0166-4328(81)90052-8 PMID: 7248055
2. Rizzolatti G, Fadiga L, Fogassi L, Gallese V. The space around us. Science. 1997; 277(5323):190–191. https://doi.org/10.1126/science.277.5323.190 PMID: 9235632
3. Serino A. Peripersonal space (PPS) as a multisensory interface between the individual and the environment, defining the space of the self. Neuroscience & Biobehavioral Reviews. 2019; 99:138–159. https://doi.org/10.1016/j.neubiorev.2019.01.016
4. Cléry J, Hamed SB. Frontier of self and impact prediction. Frontiers in Psychology. 2018; 9:1073. https://doi.org/10.3389/fpsyg.2018.01073 PMID: 29997556
5. Cléry J, Hamed SB. Functional networks for peripersonal space coding and prediction of impact to the body. In: de Vignemont F, Serino A, Wong HY, Farnè A, editors. The world at our fingertips. Oxford University Press; 2021. p. 61–79.
6. Graziano MS, Cooke DF. Parieto-frontal interactions, personal space, and defensive behavior. Neuropsychologia. 2006; 44(6):845–859. https://doi.org/10.1016/j.neuropsychologia.2005.09.009 PMID: 16277998
7. Huijsmans MK, de Haan AM, Müller BC, Dijkerman HC, van Schie HT. Knowledge of collision modulates defensive multisensory responses to looming insects in arachnophobes. Journal of Experimental Psychology: Human Perception and Performance. 2022; 48(1):1. PMID: 35073140

8. Dijkerman H, Medendorp W. Visuotactile predictive mechanisms of peripersonal space. In: de Vignemont F, Serino A, Wong HY, Farnè A, editors. The world at our fingertips: a multidisciplinary exploration of peripersonal space. Oxford University Press; 2021. p. 81–100.
9. Cléry J, Guipponi O, Odouard S, Wardak C, Hamed SB. Impact prediction by looming visual stimuli enhances tactile detection. Journal of Neuroscience. 2015; 35(10):4179–4189. https://doi.org/10.1523/JNEUROSCI.3031-14.2015 PMID: 25762665
10. 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. https://doi.org/10.1152/jn.1996.76.1.141 PMID: 8836215
11. 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. https://doi.org/10.1162/neco.2009.01-08-694 PMID: 19764874
12. Magosso E, Ursino M, di Pellegrino G, Làdavas E, Serino A. Neural bases of peri-hand space plasticity through tool-use: Insights from a combined computational–experimental approach. Neuropsychologia. 2010; 48(3):812–830. https://doi.org/10.1016/j.neuropsychologia.2009.09.037 PMID: 19835894
13. Galli G, Noel JP, Canzoneri E, Blanke O, Serino A. The wheelchair as a full-body tool extending the peripersonal space. Frontiers in Psychology. 2015; 6:639. https://doi.org/10.3389/fpsyg.2015.00639 PMID: 26042069
14. Noel JP, Blanke O, Magosso E, Serino A. Neural adaptation accounts for the dynamic resizing of peripersonal space: evidence from a psychophysical-computational approach. Journal of Neurophysiology. 2018; 119(6):2307–2333. https://doi.org/10.1152/jn.00652.2017 PMID: 29537917
15. Noel JP, Bertoni T, Terrebonne E, Pellencin E, Herbelin B, Cascio C, et al. Rapid recalibration of peripersonal space: psychophysical, electrophysiological, and neural network modeling evidence. Cerebral Cortex. 2020; 30(9):5088–5106. https://doi.org/10.1093/cercor/bhaa103 PMID: 32377673
16. Noel JP, Paredes R, Terrebonne E, Feldman JI, Woynaroski T, Cascio CJ, et al. Inflexible Updating of the Self-Other Divide During a Social Context in Autism; Psychophysical, Electrophysiological, and Neural Network Modeling Evidence. Biological Psychiatry: Cognitive Neuroscience and Neuroimaging. 2021;. https://doi.org/10.1016/j.bpsc.2021.03.013 PMID: 33845169
17. Bertoni T, Magosso E, Serino A. From statistical regularities in multisensory inputs to peripersonal space representation and body ownership: Insights from a neural network model. European Journal of Neuroscience. 2021; 53(2):611–636. https://doi.org/10.1111/ejn.14981 PMID: 32965729
18. Straka Z, Hoffmann M. Learning a Peripersonal Space Representation as a Visuo-Tactile Prediction Task. In: Lintas A, Rovetta S, Verschure PFMJ, Villa AEP, 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. Cham: Springer International Publishing; 2017. p. 101–109.
19. Roncone A, Hoffmann M, Pattacini U, Fadiga L, Metta G. Peripersonal space and margin of safety around the body: learning visuo-tactile associations in a humanoid robot with artificial skin. PloS ONE. 2016; 11(10):e0163713. https://doi.org/10.1371/journal.pone.0163713 PMID: 27711136
20. Bufacchi RJ, Liang M, Griffin LD, Iannetti GD. A geometric model of defensive peripersonal space. Journal of Neurophysiology. 2016; 115(1):218–225. https://doi.org/10.1152/jn.00691.2015 PMID: 26510762
21. Ernst MO, Banks MS. Humans integrate visual and haptic information in a statistically optimal fashion. Nature. 2002; 415(6870):429–433. https://doi.org/10.1038/415429a PMID: 11807554
22. Hillis JM, Ernst MO, Banks MS, Landy MS. Combining sensory information: mandatory fusion within, but not between, senses. Science. 2002; 298(5598):1627–1630. https://doi.org/10.1126/science.1075396 PMID: 12446912
23. Ma WJ, Beck JM, Latham PE, Pouget A. Bayesian inference with probabilistic population codes. Nature Neuroscience. 2006; 9(11):1432–1438. https://doi.org/10.1038/nn1790 PMID: 17057707
24. Van Beers RJ, Sittig AC, van der Gon JJD. Integration of proprioceptive and visual position-information: An experimentally supported model. Journal of Neurophysiology. 1999; 81(3):1355–1364. https://doi.org/10.1152/jn.1999.81.3.1355 PMID: 10085361
25. Ma WJ. Bayesian decision models: A primer. Neuron. 2019; 104(1):164–175. https://doi.org/10.1016/j.neuron.2019.09.037 PMID: 31600512
26. Duda RO, Hart PE, Stork DG. Bayesian Decision Theory. In: Pattern Classification (2nd Edition). 2nd ed. Wiley-Interscience; 2000. p. 20–83.
27. Ma WJ, Jazayeri M. Neural coding of uncertainty and probability. Annual review of Neuroscience. 2014; 37:205–220. https://doi.org/10.1146/annurev-neuro-071013-014017 PMID: 25032495
28. Colombo M, Seriès P. Bayes in the Brain—On Bayesian Modelling in Neuroscience. British Journal for the Philosophy of Science. 2012; 63(3). https://doi.org/10.1093/bjps/axr043

29. Canzoneri E, Magosso E, Serino A. Dynamic Sounds Capture the Boundaries of Peripersonal Space Representation in Humans. PLoS ONE. 2012; 7(9). https://doi.org/10.1371/journal.pone.0044306 PMID: 23028516

30. Graziano MS, Hu XT, Gross CG. Visuospatial properties of ventral premotor cortex. Journal of Neurophysiology. 1997; 77(5):2268–2292. https://doi.org/10.1152/jn.1997.77.5.2268 PMID: 9163357

31. Bufacchi RJ. Approaching threatening stimuli cause an expansion of defensive peripersonal space. Journal of Neurophysiology. 2017; 118(4):1927–1930. https://doi.org/10.1152/jn.00316.2017 PMID: 28539400

32. de Haan AM, Smit M, Van der Stigchel S, Dijkerman HC. Approaching threat modulates visuotactile interactions in peripersonal space. Experimental Brain Research. 2016; 234(7):1875–1884. https://doi.org/10.1007/s00221-016-4571-2 PMID: 26894891

33. Serino A, Noel JP, Galli G, Canzoneri E, Marmaroli P, Lissek H, et al. Body part-centered and full body-centered peripersonal space representations. Scientific reports. 2015; 5(1):1–14. https://doi.org/10.1038/srep18603 PMID: 26690698

34. de Vignemont F, Iannetti G. How many peripersonal spaces? Neuropsychologia. 2015; 70:327–334. PMID: 25448854

35. Duhamel JR, Bremmer F, Hamed SB, Graf W. Spatial invariance of visual receptive fields in parietal cortex neurons. Nature. 1997; 389(6653):845–848. https://doi.org/10.1038/39865 PMID: 9349815

36. Duhamel JR, Colby CL, Goldberg ME. Ventral intraparietal area of the macaque: congruent visual and somatic response properties. Journal of Neurophysiology. 1998; 79(1):126–136. https://doi.org/10.1152/jn.1998.79.1.126 PMID: 9425183

37. Bufacchi RJ, Iannetti GD. An action field theory of peripersonal space. Trends in Cognitive Sciences. 2018; 22(12):1076–1090. https://doi.org/10.1016/j.tics.2018.09.004 PMID: 30337061

38. Ferri F, Tajadura-Jiménez A, Väljamäe A, Vastano R, Costantini M. Emotion-inducing approaching sounds shape the boundaries of multisensory peripersonal space. Neuropsychologia. 2015; 70:468–475. https://doi.org/10.1016/j.neuropsychologia.2015.03.001 PMID: 25744869

39. Masson C, van der Westhuizen D, Noel JP, Prevost A, van Honk J, Fotopoulou A, et al. Testosterone administration in women increases the size of their peripersonal space. Experimental Brain Research. 2021; 239(5):1639–1649. https://doi.org/10.1007/s00221-021-06080-1 PMID: 33770219

40. Noel JP, Grivaz P, Marmaroli P, Lissek H, Blanke O, Serino A. Full body action remapping of peripersonal space: the case of walking. Neuropsychologia. 2015; 70:375–384. https://doi.org/10.1016/j.neuropsychologia.2014.08.030 PMID: 25193502

41. Noel JP, Pfeiffer C, Blanke O, Serino A. Peripersonal space as the space of the bodily self. Cognition. 2015; 144:49–57. https://doi.org/10.1016/j.cognition.2015.07.012 PMID: 26231086

42. Serino A, Canzoneri E, Marzolla M, Di Pellegrino G, Magosso E. Extending peripersonal space representation without tool-use: evidence from a combined behavioral-computational approach. Frontiers in Behavioral Neuroscience. 2015; 9:4. https://doi.org/10.3389/fnbeh.2015.00004 PMID: 25698947

43. Taffou M, Viaud-Delmon I. Cynophobic fear adaptively extends peri-personal space. Frontiers in Psychiatry. 2014; 5:122. https://doi.org/10.3389/fpsyt.2014.00122 PMID: 25232342

44. Posner MI. Orienting of attention. Quarterly Journal of Experimental Psychology. 1980; 32(1):3–25. https://doi.org/10.1080/00335558008248231 PMID: 7367577

45. Cléry J, Guipponi O, Odouard S, Pinède S, Wardak C, Hamed SB. The prediction of impact of a looming stimulus onto the body is subserved by multisensory integration mechanisms. Journal of Neuroscience. 2017; 37(44):10656–10670. https://doi.org/10.1523/JNEUROSCI.0610-17.2017 PMID: 28993482

46. Kandula M, Hofman D, Dijkerman HC. Visuo-tactile interactions are dependent on the predictive value of the visual stimulus. Neuropsychologia. 2015; 70:358–366. https://doi.org/10.1016/j.neuropsychologia.2014.12.008 PMID: 25498404

47. Schlack A, Sterbing-D'Angelo SJ, Hartung K, Hoffmann KP, Bremmer F. Multisensory space representations in the macaque ventral intraparietal area. Journal of Neuroscience. 2005; 25(18):4616–4625. https://doi.org/10.1523/JNEUROSCI.0455-05.2005 PMID: 15872109

48. Odegaard B, Wozny DR, Shams L. Biases in visual, auditory, and audiovisual perception of space. PLoS Computational Biology. 2015; 11(12):e1004649. https://doi.org/10.1371/journal.pcbi.1004649 PMID: 26646312

49. Makin JG, Fellows MR, Sabes PN. Learning multisensory integration and coordinate transformation via density estimation. PLoS Computational Biology. 2013; 9(4):e1003035. https://doi.org/10.1371/journal.pcbi.1003035 PMID: 23637588

Appendix D

PreCNet: Next-frame video prediction based on predictive coding

This is the accepted version of the article Straka, Z.; Svoboda, T. & Hoffmann, M. (2023), 'PreCNet: Next-frame video prediction based on predictive coding', IEEE Transactions on Neural Networks and Learning Systems. DOI: https://doi.org/10.1109/TNNLS.2023.3240857. ©IEEE

PreCNet: Next-Frame Video Prediction Based on Predictive Coding

Zdenek Straka, Tomáš Svoboda, *Member, IEEE,* and Matej Hoffmann, *Member, IEEE*

***Abstract*—Predictive coding, currently a highly influential theory in neuroscience, has not been widely adopted in machine learning yet. In this work, we transform the seminal model of Rao and Ballard (1999) into a modern deep learning framework while remaining maximally faithful to the original schema. The resulting network we propose (PreCNet) is tested on a widely used next frame video prediction benchmark, which consists of images from an urban environment recorded from a car-mounted camera, and achieves state-of-the-art performance. Performance on all measures (MSE, PSNR, SSIM) was further improved when a larger training set (2M images from BDD100k), pointing to the limitations of the KITTI training set. This work demonstrates that an architecture carefully based in a neuroscience model, without being explicitly tailored to the task at hand, can exhibit exceptional performance.**

***Index Terms*—predictive coding, deep neural networks, next frame video prediction, self-supervised learning**

I. Introduction

PREDICTING near future is a crucial ability that every agent—human, animal, or robot—needs for survival in a dynamic and complex environment. Just for safely crossing a busy road, one needs to anticipate the future position of cars, pedestrians, as well as consequences of own actions. Machines are still lagging behind in this ability. For deployment in such environments, it is necessary to overcome this gap and develop efficient methods for foreseeing the future.

One candidate approach for predicting near future is predictive coding—a popular theory from neuroscience. The basic idea is that the brain is a predictive machine which anticipates incoming sensory inputs and only the prediction errors—unpredicted components—are used for the update of an internal representation. In addition, predictive coding tackles another important aspect of perception: how to efficiently encode redundant sensory inputs [1]. Rao and Ballard proposed and implemented a hierarchical architecture [2]—which we will refer to as *predictive coding schema* (see Section III-A for details)—that explains certain important properties of the visual cortex: the presence of oriented edge/bar detectors and extra-classical receptive field effects. This schema has influenced several works on human perception and neural information processing in the brain (see e.g., [3]–[6]; for reviews [1], [7], [8]).

In this work, our goal was to remain as faithful as possible to the *predictive coding schema* but cast it into a modern deep learning framework. We thoroughly analyze how the conceptual architecture is preserved. To demonstrate the performance, we chose a widely used benchmark—next frame video prediction—for the following reasons. First, large datasets of unlabeled sequences are available and this task bears direct application potential. Second, this task is an instance of unsupervised representation learning, which is currently actively researched (e.g., [9]). Third, the complexity of the task can be scaled, for example by performing multiple frame prediction (frames are anticipated more steps ahead). On a popular next frame video prediction benchmark, our—strongly biologically grounded—network achieves state-of-the-art performance. In addition to commonly used training dataset (KITTI), we trained the model on a significantly bigger dataset which improved the performance even further.

We summarize our contributions as follows. First, in this work, the seminal predictive coding model of Rao and Ballard [2] has been cast into a modern deep learning framework, while remaining as faithful as possible to the original schema. Second, we tested our architecture (PreCNet) on a widely used next frame video prediction benchmark (KITTI with 41k images for training, Caltech Pedestrian Dataset for testing) and outperformed most of the state-of-the-art methods and achieved 2nd-3rd rank when measured with the Structural Similarity Index (SSIM)—a performance measure that should best correlate with human perception. Third, performance on all three measures (MSE, PSNR, SSIM) was significantly improved when a larger training set (BDD100k with 2M images) than usually used by the community (KITTI) was employed.

This article is structured as follows. The Related Work section overviews models inspired by predictive coding and state-of-the-art methods for video prediction. This is followed by the Architecture section where we describe our model and compare it in detail with the original Rao and Ballard schema [2] and PredNet [10]—a model for next frame video prediction inspired by predictive coding. In Section IV, we detail the datasets, performance metrics, and our experiments in next and multiple frame video prediction. This is followed by Conclusion, Discussion, and Future Work. All code and trained models used in this work are available at [11].

Z. Straka, T. Svoboda, and M. Hoffmann are with the Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, 12135 Prague, Czech Republic.
E-mail: straka.zdenek/svobodat/matej.hoffmann@fel.cvut.cz

This is the accepted version of the article Straka, Z.; Svoboda, T. & Hoffmann, M. (2023), 'PreCNet: Next-frame video prediction based on predictive coding', IEEE Transactions on Neural Networks and Learning Systems. DOI: https://doi.org/10.1109/TNNLS.2023.3240857.

II. Related Work

This section starts with a summary of predictive coding-inspired machine learning models. This is followed by an overview of state-of-the-art methods for video prediction.

A. Predictive coding models

In this section, we will focus on predictive coding-inspired machine learning models. A reader who is interested in the application in computational and theoretical neuroscience may find useful reviews [1], [8], [12] and references [2]–[6], [13]. Predictive coding, a theory originating in neuroscience, is more a general schema (with certain properties) than a concrete model. Therefore, no "correct" model of predictive coding is available to date. In this work, by predictive coding, we will understand a well defined schema proposed by Rao and Ballard [2], which was also implemented as a computational model (see Section III-A for a description of the schema). This schema, which is highly influential in neuroscience, embodies crucial ideas of the predictive coding theory.

We will relate predictive coding-inspired machine learning models to the schema by Rao and Ballard and analyze which properties of the original are preserved and which are not. A detailed comparison of our deep neural network—intended to be as faithful as possible to the Rao and Ballard schema—will be presented in a separate Section III-C1. The models with static inputs and sequences will be presented separately.

1) Models with static inputs: Song et al. [14] proposed Fast Inference Predictive Coding model (FIPC) model for image representation and classification which extends the schema by Rao and Ballard by (i) a regression procedure with fast inference during testing and (ii) a classification layer which directs representation learning to achieve discriminative features. An important part of predictive coding theory is the existence of prediction error neurons along with representational neurons (see [2], [8]). Models [15]–[17] intended for object recognition in natural images have these two distinct neural populations, however, their training is not based on the prediction error minimization used in predictive coding. A generative model by Dora et al. [18] for inferring causes underlying visual inputs does not follow the division into the error and representational neurons. However, the model is trained, in accordance with predictive coding, to minimize prediction errors. The same authors contributed to the model which extends the predictive coding approach to inference of latent visuo-tactile representations [19], used for place recognition of a biomimetic robot in a simulated environment.

2) Models with sequences as inputs: Ahmadi and Tani proposed the predictive-coding-inspired variational recurrent neural network [20] (PV-RNN). The network works in a three stage processing cycle: (i) producing prediction, (ii) backpropagating the prediction errors across the network hierarchy, (iii) updating the internal states of the network to minimize future prediction errors. The network was used for synchronous imitation between two robots—joint angles and XYZ coordinates of a hand tip were used—and for extracting latent probabilistic structure from a binary output of a simple probabilistic finite state machine. Using the same three stage predictive coding processing cycle, Choi and Tani developed a predictive multiple spatio-temporal scales recurrent neural network [21] (P-MSTRNN) for predicting binary image (36x36 pixels) sequences of human whole-body cyclic movement patterns. They also explored how the inferred internal (latent) states can be used for recognition of the movement patterns. Chalasani and Principe proposed a hierarchical linear dynamical model for feature extraction [22]. The model took inspiration from predictive coding and used higher-level predictions for inference of lower-level predictions. However, all three models do not use the division into the error and representational neurons and consequently use a different schema than Rao and Ballard [2].

Lotter et al. proposed a predictive neural network (PredNet) for next-frame video prediction [10]. The network follows the division into error and representational neurons, but the processing schema is different to the one proposed by Rao and Ballard [2] and consequently to our model (see Section III-C3 for details). Despite the architectural differences from the schema by Rao and Ballard, the network could mimic certain features of biological neurons and perception [23].

B. Video prediction models

Video prediction is an important task in computer vision with a long history. A sequence of images is given and one or multiple following images are predicted (i.e., next and multiple frame video prediction task respectively). As prediction of the next sensory input is inherent to predictive coding, next frame video prediction provides a natural use case to benchmark the performance of our neural network architecture. Therefore, we will focus predominantly on a brief review of recent work with state-of-the-art performance on next frame video prediction and—wherever feasible—we will quantitatively compare the performance (see Section IV-C4).

Many of the methods for video prediction produce blurred predictions. As blurriness is undesirable, Matthieu et al. [9] proposed a gradient difference loss function which is minimized when the gradient of the actual and predicted image is the same. This loss function was then combined with adversarial learning. Byeon et al. [24] showed with their LSTM-based architecture that direct connection of each predicted pixel with the whole available past context led to decreasing prediction uncertainty on pixel level and therefore also reduced blurriness. Reda et al. [25] suggested that blurriness is amplified by using datasets with lack of large motion and small resolution. Therefore, they used video games (GTA-V and Battlefield-1) for generation of a large high-resolution dataset with large enough motion (testing was performed on natural sequences). The dataset was then used for training of a model which combines a kernel-based approach with usage of optical flow. In addition to optical flow estimation, a model by Lu et al. [26] used pixel generation and adversarial training. Gao et al. [27] proposed a model which performed generation of the future frames in two steps. Firstly, a flow predictor was used for warping the non-occluded regions. Then, the occluded regions were in-painted by a separate network. A method by Liu et al. [28] did not use optical flow directly,

however, a deep network was trained to synthesize a future frame by flowing pixel values from the given video frames. This self-supervised method was also used for interpolation. Similarly to Gao et al. [27], Hao et al. [29] proposed a two-stage architecture. However, the input of a network contained, in addition, sparse motion trajectories (automatically extracted for video prediction). First, the network produced a warped image that respected the given motion trajectories. In the second stage, occluded parts of the image were hallucinated and color change was compensated.

Villegas et al. [30] introduced a model which first performed human pose detection and its future evolution. Then, the predicted human poses were used for future frames generation.

Finn et al. [31] proposed a model that next to visual inputs takes actions of the robot into account. This action-conditioned model learned to anticipate pixel motions relatively to the previous frame.

A Conditionally Reversible Network (CrevNet) proposed by Yu et al. [32] uses a bijective two-way autoencoder, based on convolutional networks, for encoding and decoding input frames. Feature maps obtained from the autoencoder are then used as an input to a ConvRNN-based predictor. The transformed feature maps by the predictor are then decoded by the autoencoder and outputted as predicted frames. Chang et al. proposed Information Preserving Spatiotemporal Predictive Model [33] (IPRNN) which used skip connections from encoders to decoders. As a decoder has access to information from an encoder, the information loss is reduced. The model used stacked spatiotemporal gated recurrent units which took the encoded states as input. Yuan et al. integrated the attention mechanism into a convolutional LSTM network [34]. This model was further extended by pixel restoration of the input images to the predictions and denoted as Deep Pixel Restoration AttConvLSTM (DPRAConvLSTM) model. Attention mechanism was also effectively used for human-skeleton motion prediction [35], [36].

Some other state-of-the-art architectures are based on generative adversarial networks (GANs). The GAN by Kwon and Park [37] is trained to anticipate both future and past frames. The GAN proposed by Liang et al. [38] is trained to consistently predict future frames and pixel-wise flows using a dual learning mechanism. Vondrick et al. [39] proposed GAN for generation of image sequences which unravels foreground from the background of the images. A video prediction network proposed by Jin et al. [40] integrates generative adversarial learning with usage of spatial and temporal wavelet analysis modules.

The stochastic nature of natural video sequences makes it impossible to predict the future sequence perfectly. Models such as [41]–[43] attempt to deal with that by generating multiple possible futures. The objective is to predict frame sequences which are: (i) diverse, (ii) perceptually realistic, and (iii) a plausible continuation of the given input sequence or image [41]. Therefore, this task is different from deterministic video frame prediction whereby the model is intended to produce only a single frame or sequence best fitting the actual future.

Some of the mentioned works [10], [32], [38], [39] also demonstrated that the representations which were learned during next frames video prediction training could be used for supervised learning tasks (e.g., human action recognition).

III. Architecture

This section starts with a description of the *predictive coding schema* which was proposed by Rao and Ballard [2]. This is followed by a detailed description of our model. The section is closed by a comparison of our model with related models: (i) a hierarchical network for predictive coding proposed by Rao and Ballard, (ii) PredNet – a deep network for next frame video prediction inspired by predictive coding.

A. Predictive coding schema

Motivated by crucial properties of the visual cortex, Rao and Ballard have proposed a hierarchical *predictive coding schema* with its implementation [2]. According to this schema, throughout the hierarchy of visual processing, "feedback connections from a higher- to a lower-order visual cortical area carry predictions of lower-level neural activities, whereas the feedforward connections carry the residual errors between the predictions and the actual lower-level activities" [2]. The residual errors are used to reduce the prediction error in the following moment (see Fig. 1, **(b)**).

This schema was directly turned into a computational model in [2] (see Fig. 1, **(a)**). The feedback connection from higher-level to lower-level Predictive Estimator (PE) carries the top-down prediction $\mathbf{r}^{td}$ of the lower-level PE activity $\mathbf{r}$. The residual error $\mathbf{r}-\mathbf{r}^{td}$ is sent back via feedforward connections to the higher-level PE. The same error with opposite sign, $\mathbf{r}^{td}-\mathbf{r}$, affects the following PE activity $\mathbf{r}$ (see Fig. 1, **(a)**). The bottom-level PE produces a prediction of the visual input.

Drawing on the *predictive coding schema*, we propose the *Predictive Coding Network (PreCNet)* (see Fig. 1, **(c)**). In contrast with the model by Rao and Ballard (compare parts **(a), (c)** of Fig. 1), PreCNet uses a modern deep learning framework (see Section III-B for details of PreCNet architecture and Section III-C1 for a more detailed comparison of both models). This has enabled us to create a model based on the *predictive coding schema* with state-of-the-art performance, as demonstrated on the next-frame video prediction benchmark.

B. Description of PreCNet model (ours)

The structure, computation of prediction and states, and training of the model is detailed below.

1) Structure of the model: The model, shown in Fig. 2, consists of $N+1$ hierarchically organized modules[1]. A module $i \in \{0, 1, \ldots, N\}$ consists of the following components:

- A **representation layer** is a convolutional LSTM (convLSTM$_i$) layer (see [44], [45]) with output state R_i (alternatively[2] $\mathbf{r}$). The convLSTM followed dynamics

[1]The model is the same as in Fig. 1, **(c)**. However, in order to enable direct comparison with Rao and Ballard model, it was redrawn in a different arrangement for Fig. 1, **(b)**. The PE from the model of [2] is not equivalent to the "Module" in Fig. 2. See Fig. 3, **(b)**, **(e)** for a comparison.

[2]For representation layer states we used both small $\mathbf{r}$ and capital letter R. Small $\mathbf{r}$ corresponds to formalism from [2], capital R was used in [10].

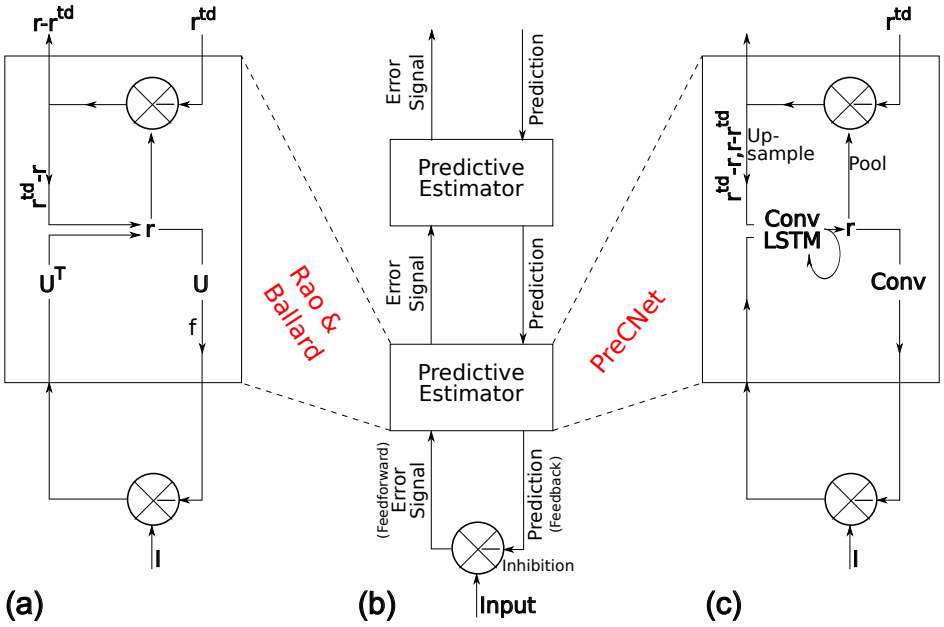

Fig. 1. **Comparison of the hierarchical network for predictive coding by Rao and Ballard and our PreCNet. (a) Components of a Predictive Estimator (PE) module of the model by Rao and Ballard**, composed of feedforward neurons encoding the synaptic weights $\mathbf{U}^T$, neurons whose responses $\mathbf{r}$ maintain the current estimate of the input signal, feedback neurons encoding $\mathbf{U}$ and conveying the prediction $f(\mathbf{Ur})$ to the lower level, and error-detecting neurons computing the difference $(\mathbf{r} - \mathbf{r}^{td})$ between the current estimate $\mathbf{r}$ and its top-down prediction $\mathbf{r}^{td}$ from a higher level. **(b) General architecture of the hierarchical predictive coding model.** At each hierarchical level, feedback pathways carry predictions of neural activity at the lower level, whereas feedforward pathways carry residual errors between the predictions and actual neural activity. These errors are used by the PE at each level to correct its current estimate of the input signal and generate the next prediction. **(c) Components of a PE module of PreCNet architecture** (see Section III-C1). Figures **(a)** and **(b)** redrawn from [2], their captions with minor modification from [2].

from commonly used "No peepholes" LSTM variant [46] (see *Supplementary materials – convLSTM* for details). Technically, it consists of two convolutional LSTM layers (convLSTM$_i^{up/down}$) which share hidden and cell states (R_i, C_i) but differ in the input (E_i vs. E_{i+1}). The input, forget, and output gates use hard sigmoid as an activation function. During calculation of the final (hidden) and cell states, hyperbolic tangent is used.

- An **error representation** consists of the Rectified Linear Units (ReLU) whose input is obtained by merging errors $PREDICTION - ACTUAL\ STATE$ and $ACTUAL\ STATE - PREDICTION$. The state of the error representation is denoted as E_i.
- A **decoding layer** is a convolutional (conv$_i$) layer with output state $\hat{A}_i$. It uses ReLU as an activation function.
- An **upsample layer**, which uses nearest-neighbor method, upscales its input by factor 2. This layer is not present in the module 0.
- A **max-pooling layer** which downscales its input by a factor 2. This layer is not present in the module 0.

2) Computation of the prediction and states: In every time step, PreCNet outputs a prediction of the incoming image. The error of the prediction is then used for the update of the states (see also Fig. 4). The computation in every time step can be divided into two phases:

1) **Prediction phase**. The information flow goes iteratively from a higher to a lower module. At the end of this phase (at Module 0), the prediction of the incoming input image $\hat{A}_0$ is outputted.
2) **Correction phase**. In this phase, the information flow goes iteratively up. The error between the prediction and actual input is propagated upward.

In a nutshell, a representational layer (with state R_i) represents a prediction of the image I ($i = 0$) or a pooled convLSTM state R_{i-1} from the module bellow ($i > 0$). The decoding layer transforms the representation R_i into the prediction $\hat{A}_i$. The error representation units E_i then depend on the error of the prediction $\hat{A}_i$ (difference between the prediction $\hat{A}_i$ and the image I or the pooled state R_{i-1}). The computation is completely described in Alg. 1.

3) Training of the model: The model is trained by minimizing weighted prediction errors through the time and hierarchy [10]. The loss function is defined as

$$L_{train} = \sum_{m=1}^{M} L_{seq}(m), \tag{1}$$

$$L_{seq}(j) = \sum_{t=1}^{l_s} \mu_t \sum_{l=0}^{N} \frac{\lambda_l}{n_l} \sum_{i=1}^{n_l} E_l^t(i), \tag{2}$$

where $L_{seq}(m)$ is loss of the m^{th} sequence, $E_l^t(i)$ is the error of the i^{th} unit in the module l at time t, M is a number of image sequences, l_s is a length of a sequence, $N + 1$ is a number of modules, μ_t, λ_l are time and module weighting factors, n_l is the number of error units in the l^{th} module. The mini-batch gradient descent was used for the minimization.

This is the accepted version of the article Straka, Z.; Svoboda, T. & Hoffmann, M. (2023), 'PreCNet: Next-frame video prediction based on predictive coding', IEEE Transactions on Neural Networks and Learning Systems. DOI: https://doi.org/10.1109/TNNLS.2023.3240857. ©IEEE

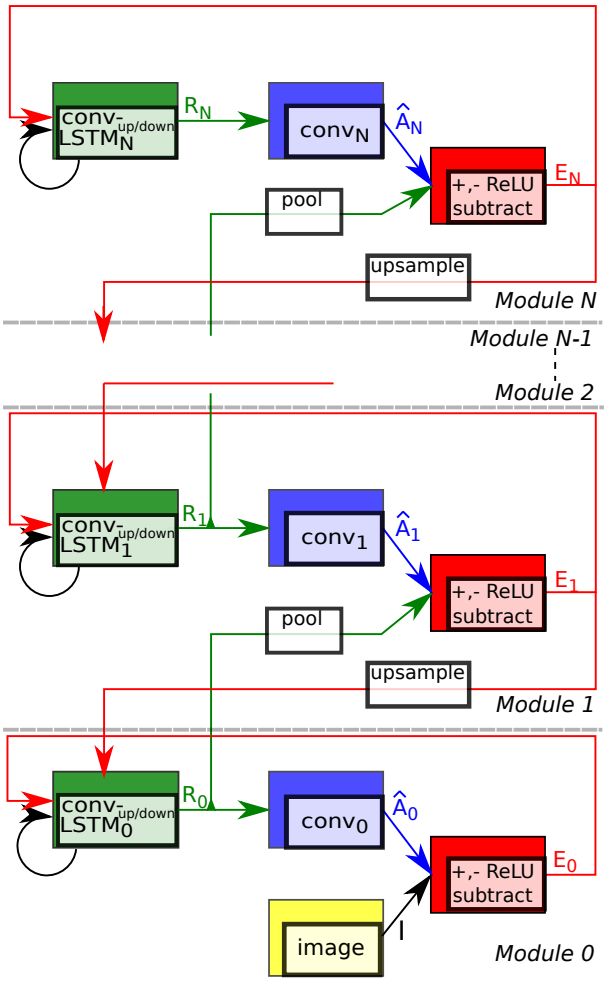

Fig. 2. **Modular architecture of PreCNet.** The highest module misses connections upwards. Main parts of each module are a representation layer (green), decoding layer (blue) and error representations (red). See the text and Alg.1 for more details.

Algorithm 1 Calculate PreCNet states at time t, assume $N > 0$. Merging of states A and B is denoted by putting them into curly brackets {A, B}. See *Supplementary materials – convLSTM* for a detailed convLSTM description.

Require: Image I^t, previous $(t-1)$ hidden and cell states R_l^{t-1}, C_l^{t-1} of the representation layers $l \in \{0, 1, \ldots, N\}$, previous error state E_N^{t-1} of the (top) module N, maximum pixel value pix^{max}. For $t = 1$, the previous states R_l^0, C_l^0, E_N^0 are initialized to zero.

for $l = N, N-1, \ldots, 0$ {Iterate top-down through the modules} **do**
  if $l == N$ {Update the states in the top module} **then**
    $R_l^t, C_l^t \leftarrow \text{convLSTM}_l^{down}(R_l^{t-1}, C_l^{t-1}, E_l^{t-1})$
    $\hat{A}_l^t \leftarrow \text{conv}_l(R_l^t)$
    $E_l^t \leftarrow \text{ReLU}(\{\hat{A}_l^t - \text{pool}(R_{l-1}^{t-1}), \text{pool}(R_{l-1}^{t-1}) - \hat{A}_l^t\})$
  if $l \neq N$ **and** $l \neq 0$ {Update the states in the "middle" module l} **then**
    $R_l^t, C_l^t \leftarrow \text{convLSTM}_l^{down}(R_l^{t-1}, C_l^{t-1}, \text{upsample}(E_{l+1}^t))$
    $\hat{A}_l^t \leftarrow \text{conv}_l(R_l^t)$
    $E_l^t \leftarrow \text{ReLU}(\{\hat{A}_l^t - \text{pool}(R_{l-1}^{t-1}), \text{pool}(R_{l-1}^{t-1}) - \hat{A}_l^t\})$
  if $l == 0$ {Update the states in the bottom module} **then**
    $R_l^t, C_l^t \leftarrow \text{convLSTM}_l^{down}(R_l^{t-1}, C_l^{t-1}, \text{upsample}(E_{l+1}^t))$
    $\hat{A}_l^t \leftarrow \min(\text{conv}_l(R_l^t), pix^{max})$
    $E_l^t \leftarrow \text{ReLU}(\{\hat{A}_l^t - I^t, I^t - \hat{A}_l^t\})$
for $l = 0, 1, \ldots, N$ {Iterate bottom-up through the modules} **do**
  if $l == 0$ **then**
    $R_l^t, C_l^t \leftarrow \text{convLSTM}_l^{up}(R_l^t, C_l^t, E_l^t)$
  if $l \neq 0$ **and** $l \neq N$ **then**
    $E_l^t \leftarrow \text{ReLU}(\{\hat{A}_l^t - \text{pool}(R_{l-1}^t), \text{pool}(R_{l-1}^t) - \hat{A}_l^t\})$
    $R_l^t, C_l^t \leftarrow \text{convLSTM}_l^{up}(R_l^t, C_l^t, E_l^t)$
  if $l == N$ **then**
    $E_l^t \leftarrow \text{ReLU}(\{\hat{A}_l^t - \text{pool}(R_{l-1}^t), \text{pool}(R_{l-1}^t) - \hat{A}_l^t\})$

C. Comparison of PreCNet with other models

We will compare our model with the *predictive coding schema* [2], the Fast Inference Predictive Coding model (FIPC) [14], and PredNet [10].

1) Comparison of PreCNet and Rao and Ballard model: PreCNet uses the same schema as the model by Rao and Ballard (see Fig. 1 and Section III-A). However, as PreCNet is couched in a modern deep learning framework and uses video sequences as inputs, there are inevitably some differences. The crucial differences are:

- Dynamic vs. static inputs. In contrast with PreCNet and image sequences as inputs, the model by Rao and Ballard takes static images as inputs. An extension to next frame video prediction should be possible [47][3], but has not been completely demonstrated (in [48], a model with only one level of hierarchy is employed). These recurrent connections resemble the recurrent connections inside PreCNet representation (convLSTM) layer.
- Different building blocks. PreCNet, in contrast to Rao and Ballard model, uses modern deep learning blocks – convolutional and convLSTM layers. In addition, the error representation of PreCNet consists of merged positive, $PREDICTION - ACTUAL\ STATE$, and negative, $ACTUAL\ STATE - PREDICTION$, error populations [10]. These two populations are also used in the model of Rao and Ballard, however, they are not merged and are used separately. In contrast to PreCNet, ReLU is not applied to the error populations.
- Different update of representation states. Representation layer states of the Rao and Ballard model are deter-

[3]By using recurrent transformation of the representation layer states $\hat{\mathbf{r}}(t+1) = f(V\mathbf{r}(t))$, where $\hat{\mathbf{r}}(t+1)$ is the prediction of the next state $\mathbf{r}(t+1)$ made at time t, f is a nonlinear function, and V are synaptic recurrent weights.

mined by a first-order differential equation. The states are updated until they converge. In contrast, PreCNet's representation layer states are calculated by convLSTM. To update the representation states $\mathbf{r}$ of the model by Rao and Ballard, the "bottom-up" difference between the prediction of the PE and the actual input ($I - f(U\mathbf{r})$ in Fig. 1, **(a)**) followed by fully connected layer and the "top-down" difference between the predicted state by the higher PE and the actual state of the PE ($\mathbf{r}^{td} - \mathbf{r}$ in Fig. 1, **(a)**) are used simultaneously. PreCNet also uses both differences for computation of the new representation states $\mathbf{r}$, however, not simultaneously; one difference is used by the convLSTM^{down} during the prediction phase, the second is used by the convLSTM^{up} during the correction phase (notice that the convLSTM^{down} and convLSTM^{up} share cell and hidden unit states).

- One vs. mutliple PEs on one level. There are multiple PEs in one level of the model by Rao and Ballard. Higher level PEs progressively operate on bigger spatial areas than the lower level PEs. PreCNet has one PE in each level of the hierarchy.
- Intensity of interaction between the Predictive Estimators (PEs). Each PE of PreCNet is updated just two times during one time step (one input image). Once during the Prediction (top-down) phase and once during the Correction (bottom-up) phase. This means that each PE interact with its neighbour just two times during one time step. On the contrary, the PEs of the model by Rao and Ballard interact with each other many times (until their representation states converge) during one time step. As PreCNet uses the deep learning approach, which is more computationally demanding, such intensive interaction between the PEs is not possible.
- Minimizing error in all levels vs. only the bottom level error. Errors in all levels of the model by Rao and Ballard are minimized. However, PreCNet has achieved better results when only the bottom level error—the difference between the predicted and the actual image—was minimized (see the setting of parameter λ_i in Section IV-C2).

2) Comparison of PreCNet and FIPC: As FIPC [14] is mainly an extension of the original Rao and Ballard model by a procedure for regression mapping with fast inference at test time, it shares many properties with the Rao and Ballard model. The most important differences between PreCNet and FIPC are:

- Different building blocks. Main building blocks of PreCNet are convolutional and convolutional LSTM networks. FIPC main basic building blocks, similarly to Rao and Ballard model, are simple feedforward networks. This might be limiting for usage on large-scale images and video sequences.
- Dynamic vs. static inputs. PreCNet takes image sequences as inputs and predicts next frames. FIPC, identically to Rao and Ballard model, works with static images and is trained for their classification and feature representation.
- Fast inference at test time. During testing, the trained network works as a feedforward network (a subset of weights is used) with class labels as outputs. Therefore, during test time the network does not follow the *predictive coding schema*.
- Intensity of interaction between the Predictive Estimators (PEs) during training. FIPC Predictive Estimators, similarly to Rao and Ballard model, interact with each other many times during one training time step. For PreCNet, it is only two times (see Section III-C1).
- Classification layer. FIPC added to the *predictive coding schema* a classification layer which helps to learn more discriminative features for a given classification task. On the other hand, PreCNet is completely self-supervised.

3) Comparison of PreCNet and PredNet: PredNet, a state-of-the-art deep network for next frame video prediction [10], is also inspired by the model by Rao and Ballard. PredNet and PreCNet (which we propose) are similar in these aspects:

- Building blocks: error representations, convolutional, and convolutional LSTM networks.
- Training procedure. For the next frame video prediction task, most training parameters, such as input sequence length and batch size, of PreCNet are taken from PredNet[4].
- Number of trainable parameters. For training on the KITTI dataset, PreCNet had approximately 7.6M and PredNet 6.9M trainable parameters. We tested also PreCNet-small with 0.8M parameters (see Table III for results).

However, there are two crucial properties in which PredNet departs from the *predictive coding schema* (see Fig. 3, **(a), (c), (d)**):

- According to the *predictive coding schema*, except for the bottom Predictive Estimator (PE), each PE outputs a prediction of the next lower level PE activity $\mathbf{r}$ (representation layer state). See Section III-A.
- No direct connection between two neighboring PE activities $\mathbf{r}_i$ and $\mathbf{r}_{i-1}$ (representation layer states $\mathbf{R}_i$ and $\mathbf{R}_{i-1}$ in formalism of [10]).

Instead, to remain faithful to the *predictive coding schema*, the building blocks of PreCNet were connected in a significantly different way (see Fig. 3 for comparison). These modifications have led to considerably better performance of PreCNet in comparison with PredNet (see Section IV-C4).

IV. Experiments

In this section, the datasets and performance measures are introduced, followed by experiments on next frame and multiple frame video prediction. Trained models and code needed for replication of all the results presented in the paper (dataset preprocessing, model training and evaluation) are available on a **GitHub repository** [11].

[4]The motivation was two-fold. Firstly, we wanted to make it clear that the significant improvement of PreCNet over PredNet is not caused by better choice of training parameters. Secondly, few trials with other parameter values that we tried did not lead to significantly better results.

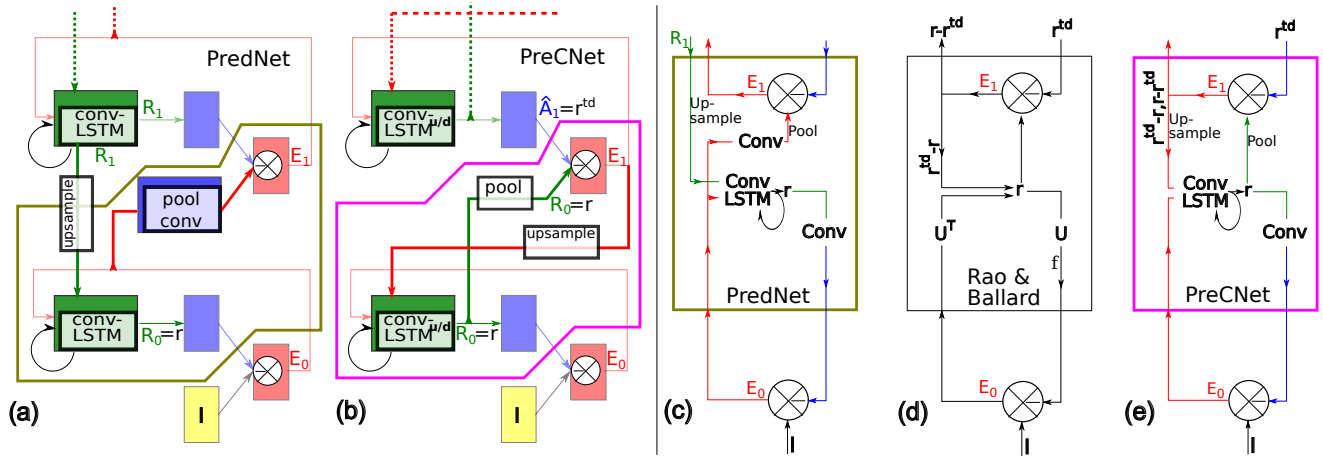

Fig. 3. **Comparison of PredNet and PreCNet.** In **(a), (b)**, the differences (connections between the blocks, some building blocks) are highlighted. In **(c), (d), (e)**, there is a comparison of the Predictive Estimators (PEs) of PredNet, PrecNet and the model by Rao and Ballard. Notice that the input from above in **(d), (e)** – prediction $\mathbf{r}^{td}$ of $\mathbf{r}$ – is compared with the representation state $\mathbf{r}$ and the error is used for update of the $\mathbf{r}$. The corresponding upper input (blue) of the PredNet is a different entity; it is not related to $\mathbf{r}$ and is also compared with a different entity (Conv(E_0)). There is also one more input from above – representation layer state from above – which goes directly into ConvLSTM block of PredNet. PreCNet (see **(e)**) has overcome these differences and follows the same predictive coding scheme as the model by Rao and Ballard. Notice the correspondence of the olive, purple polygons (**(a), (b)**) and the PEs of PredNet and PreCNet (the rectangles in **(c), (e)**). In order to enhance comprehensibility, some of the labels from **(a), (b)** were added to **(c), (d), (e)** and v.v. See *Supplementary materials – Schema transformation* to check the correspondence between both (**(a), (b)** and **(c), (e)**) ways of visualization.

A. Datasets

All datasets used are visual sequences obtained from a car mounted camera. These scenes include fast movements of complex objects (e.g. cars, pedestrians), new objects coming unexpectedly to the scene, as well as movement of the urban background.

For training, we used two different datasets; KITTI [49] and BDD100K [50]. For evaluation, we used Caltech Pedestrian Dataset [51], [52], employing Piotr's Computer Vision Matlab Toolbox [53] during preprocessing. Using of Caltech Pedestrian Dataset for establishing performance enables direct comparison of the models from both training variants.

- **KITTI dataset and its preprocessing:** We followed the preprocessing procedure from [10]. The frames were center-cropped and resized with bicubic method[5] to 128 by 160 pixels size (see the repository for code). We also followed the division categories "city", "residential" and "road" of the KITTI dataset to training (57 recording sessions, approx. 41K of frames) and validation parts in the same way as in [10]. The dataset has 10 fps frame rate.
- **Caltech Pedestrian Dataset and its preprocessing:** Frames were preprocessed in the same way as the frames of KITTI dataset (see above). Videos were downsampled from 30 fps to 10 fps (every 3rd frame was taken). As this dataset was used only for evaluation of the performance, only testing parts (set06-set10) were used (approx. 41K of frames).
- **BDD100K and its preprocessing:** The preprocessing of the dataset was analogous to the preprocessing of Caltech Pedestrian Dataset, including reducing frame rate from 30 to 10 fps. As the size of the whole dataset is very large (roughly 40M frames if 10 fps is used), we had to randomly choose training and validation subsets of the dataset—see the repository for details and chosen videos. We created two variants of the training dataset; a big one with roughly 2M frames (5000 recording sessions) and a small one with similar size like KITTI training dataset (approx. 41K frames, 105 recording sessions). As a validation dataset, we randomly selected a subset of the validation part of BDD100K with approx. 9K frames.

[5]We do not know which resizing method was originally used by Lotter et al. [10].

B. Performance measures

For comparison of a predicted with the actual frame, we use standard measures: Mean Square Error (MSE), Peak Signal-to-Noise Ratio (PSNR), and Structural Similarity Index (SSIM) [54]. MSE is a simple measure whose low values indicate high similarity between frames. PSNR is a related measure to MSE whose value is desired to be as high as possible. Significant limitation of these two is that their evaluation of similarity between two images does not correlate very well with human judgment (e.g., [55], [56]). SSIM was created to be more correlated with human perception. SSIM values are bounded to $[-1, 1]$ and higher value signifies higher similarity.

C. Next frame video prediction

Firstly, the settings of experiments and parameters will be described. This is followed by Quantitative results and Qualitative analysis. Results, achieved by PreCNet, presented in this subsection can be generated by publicly available code [11]. Summary and details of the network parameters and training are in *Supplementary materials – 1 Network and training parameters summary and details*.

TABLE I
NETWORK PARAMETERS SUMMARY. SEE TEXT FOR A DESCRIPTION.

		$conv_i$		$convLSTM_i^{up/down}$	
module	weight λ_i	#chan.	filter size	#chan.	filter size
i=0	1	3	3	60	3
i=1	0	60	3	120	3
i=2	0	120	3	240	3

1) Experimental settings: We performed experiments with two settings. In both, the performance of trained models was measured using *Caltech Pedestrian Dataset* (see Section IV-A) which is commonly used for evaluating next frame video prediction task. This also enabled direct comparison of training on both datasets. The training was done on:

- **KITTI dataset.** This setting (i.e., KITTI for training, Caltech Pedestrian Dataset for evaluation) is popular for evaluation of next frame video prediction task and enables good comparison with other state of the art methods.
- **BDD100K dataset.** Randomly chosen subset of the dataset (approx. 2M of frames) was used. The training dataset is significantly larger than KITTI dataset which enables to avoid overfitting. We also performed training on smaller BDD100K subset with roughly same size as KITTI training dataset.

2) Network parameters: Main parameters of the network are summarized in Table I. In the Table, the parameters of each module in the hierarchy are described in a row. Module weights are in the second column. The following columns contain the number of channels #chan. (layer size) and filter sizes of decoding (conv) and representation (convLSTM) layers. For a detailed explanation see Section III-B.

For choosing a suitable number of hierarchical modules, layer sizes (number of channels), and module weight factors ($\lambda_i, i \in \{0, 1, 2\}$), KITTI dataset was used for training. We performed a manual heuristic parameter search to minimize mean absolute error (between the predicted and actual frames) on validation set[6]. Padding was used to preserve the size in all convolutional layers (including convLSTM). Values of the pixels of the input frames were divided by 255 to make them in the range $[0, 1]$. The filter sizes were taken from [10] (for explanation of this choice, see Section IV-C3).

To better understand how the number of trainable parameters affects the performance and better comparison with PredNet, we proposed the same model but changed the number of channels in the modules from 60, 120, 240 to 20, 40, and 80 respectively (*PreCNet-small*). The number of parameters was reduced from 7.6M to 0.8M. Moreover, we simplified the architecture by (i) replacing all pairs of convolutional LSTMs with shared hidden and cell states—convLSTM$_i^{up}$, convLSTM$_i^{down}$— by single convolutional LSTMs – convLSTM$_i$ (*PreCNet-single-LSTMs*), and (ii) by simplifying error blocks *ReLU({PREDICTION-ACTUAL, ACTUAL-PREDICTION})* (see Alg. 1) to residual errors PREDICTION-ACTUAL only (*PreCNet-residual-error*). It was also necessary to modify the sequence loss (2) by putting the error values $E_l^t(i)$ into absolute value.

3) Training parameters: Except for training length and learning rate, all the values of the training parameters were same as in [10] (see Section III-C3 for the explanation). The network was trained on input sequences with length $l_s = 10$. In the sequences used for training and validation, a frame was generally present in more sequences, meaning that the sequences overlap. During learning, the error related to the first predicted input is ignored ($\mu_{t=1} = 0$), since the first prediction is produced before seeing any input frame. Prediction errors related to the following time steps are equally weighted ($\mu_t = \frac{1}{l_s - 1}$, for $t \in \{2, .., l_s\}$).

In each epoch, 500 sequences from the training set were randomly selected to form batches of size 4 and used for weight updates. For validation, 100 randomly selected sequences from the validation set were used in each epoch. We used Adam [57] as an optimization method for stochastic gradient descent on the training loss (1). The values of the Adam parameters β_1, β_2 were set to their default values ($\beta_1 = 0.9$, $\beta_2 = 0.999$).

Training parameters for training on both datasets were very similar except for number of training epochs and learning rate setting. For the KITTI and BDD100K training, the learning consists of 1000 and 10000 epochs, respectively. Learning rate was set to 0.001 and 0.0005 for first 900, 9900 epochs, respectively[7]. Then it was decreased to 0.0001 for last 100 epochs. As the BDD100K training set is significantly larger than KITTI training set, the training was longer for BDD100K. The choice of the length of the training and learning rate was based on evolution of validation loss and limited computational resources. It means that validation loss still slightly decreased at the final epochs, however, the benefit was not so significant to continue training and use (limited) computational resources.

4) Quantitative results: For a quantitative analysis of the performance of the model, we used a standard procedure and measures for evaluating the next frame video prediction. The network obtained a sequence (from Caltech Pedestrian Dataset) of length 10 and then predicted the next frame (see Fig. 4 for details). Contrary to training and validating sequences, there was no overlap between the two testing sequences of length 11. This frame is compared to the actual frame using MSE, PSNR and SSIM (see Section IV-B). The overall value of each measure is then obtained as a mean of the calculated values for each predicted frame.

We performed 10 training repetitions on **KITTI dataset** (see Section IV-C1). The results are summarized in Table II. The results show that the learning is stable. Moreover, we carried out one training repetition of PreCNet-small, PreCNet-residual-error and PreCNet-single-LSTMs.

We took the best model of 10 repetitions (according to SSIM) and compared it with state-of-the-art methods (see Table III). In the Table, the methods are sorted according to their SSIM values. If not stated otherwise, a network

[6]If $\lambda_0 = 1, \lambda_{1,2} = 0$ then the mean absolute error between the predicted and actual frames corresponds to 2*loss value (2). This is a consequence of division of error representation to negative and positive parts and using of ReLU. For non zero $\lambda_{1,2}$, this does not hold.

[7]Learning rate setting 0.001 for BDD100K training led in two of four cases to rapid increase of training loss in later stages of training. Therefore, the learning rate was changed to 0.0005.

This is the accepted version of the article Straka, Z.; Svoboda, T. & Hoffmann, M. (2023), 'PreCNet: Next-frame video prediction based on predictive coding', IEEE Transactions on Neural Networks and Learning Systems. DOI: https://doi.org/10.1109/TNNLS.2023.3240857. ©IEEE

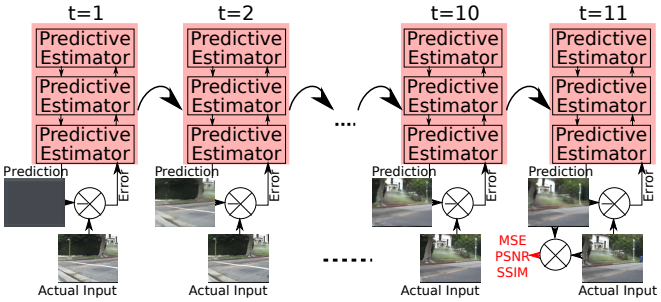

Fig. 4. **Next frame video prediction evaluation schema.** In each time step PreCNet outputs next frame prediction. The predicted error is used for update of the network states. After inputting 10 frames (time step $t = 11$), the predicted frame is compared—using MSE, PSNR, SSIM—with the actual input. This schema was used for quantitative and qualitative analysis of Next frame video prediction (see Section IV-C).

TABLE II
PERFORMANCE SUMMARY OF 10 TRAINING REPETITIONS ON KITTI DATASET. CALTECH PEDESTRIAN DATASET WAS USED FOR CALCULATION OF THE VALUES. SEE SECTION IV-C4 FOR DETAILS.

	MSE	PSNR	SSIM
best value	0.00205	28.4	0.929
worst value	0.00220	28.1	0.928
median	0.00208	28.4	0.928

received ten input images and predicted the next one which was used during performance evaluation. Unless otherwise stated, the values were taken from the original articles. Values for BeyondMSE were taken from [38]; values for DVF and CtrlGen were taken from [27]. Values for PredNet were taken from [24], because in [10] the values were averaged over nine (2-10) time steps. Values for "PreCNet 7 input frames" (see Table IV), RC-GAN, DPRAConvLSTM and Lu et al. model were calculated after only seven, four, three and two input images (not ten), respectively. However, "PreCNet 7 input frames" and RC-GAN had better performance in this case than for input sequences of length ten. If this is true also for DPRAConvLSTM and Lu et al. model is not known. The number of parameters for DM-GAN and PredNet were taken from [24].

PreCNet achieved 2nd-3rd position in SSIM. In MSE and PSNR, it was outperformed by four and seven other methods, respectively. The number of trainable parameters—for the models where it is available—is similar for all except DM-GAN (113M) and PreCNet-small (0.8M). PreCNet with a small number of parameters still had comparable performance to other models and outperformed PredNet. Replacement of the pairs of $\text{convLSTM}_i^{up/down}$ by a single convLSTM_i and simplification of error blocks degraded the performance only slightly.

Moreover, we took the best trained model and investigated its performance for shorter input testing sequences. The results are shown in Table IV. The network received input sequences with the given input length and predicted the next frame. Except the input sequence length, the experiential setting and the PreCNet network are the same as in Tab. III (the values for input length 10 are the same as in Table III). Copy last MSE is not the same for all input lengths because the test set was

TABLE III
NEXT FRAME VIDEO PREDICTION PERFORMANCE ON CALTECH PEDESTRIAN DATASET AFTER TRAINING ON KITTI DATASET. SEE TEXT FOR DETAILS.

	Caltech Pedestrian Dataset			
Method	MSE	PSNR	SSIM	#param
Copy last frame	0.00795	23.2	0.779	-
BeyondMSE [9]	0.00326	-	0.881	-
DVF [28]	-	26.2	0.897	-
DM-GAN [38]	0.00241	-	0.899	113M
CtrlGen [29]	-	26.5	0.900	-
PredNet [10]	0.00242	27.6	0.905	6.9M
Lu et al. [26]	0.00188	28.7	0.913	3.9M
RC-GAN [37]	0.00161	29.2	0.919	-
ContextVP [24]	0.00194	28.7	0.921	8.6M
DPG [27]	-	28.2	0.923	-
CrevNet [32]	-	29.3	0.925	-
Jin et al. [40]	-	29.1	0.927	7.6M
PreCNet (ours)	0.00205	28.4	0.929	7.6M
PreCNet 7 input frames (ours)	0.00202	28.5	0.930	7.6M
DPRAConvLSTM [34]	-	30.2	0.930	-
IPRNN [33]	**0.00097**	**31.0**	**0.955**	-
PreCNet-small (ours)	0.00220	28.0	0.919	0.8M
PreCNet-single-LSTMs (ours)	0.00209	28.3	0.926	7.0M
PreCNet-residual-error (ours)	0.00212	28.2	0.927	5.6M

split into non-overlapping sequences with different lengths. The performance was significantly worse for input length 3

TABLE IV
NEXT FRAME PREDICTION PERFORMANCE FOR DIFFERENT LENGTH OF INPUT SEQUENCE. SEE TEXT FOR A DESCRIPTION.

	Caltech Pedestrian Dataset			
Input length	MSE	PSNR	SSIM	Copy last MSE
3	0.00216	28.1	0.924	0.00796
4	0.00208	28.4	0.928	0.00794
5	0.00203	28.5	0.929	0.00795
6	0.00204	28.5	0.930	0.00799
7	0.00202	28.5	0.930	0.00798
8	0.00203	28.5	0.930	0.00794
9	0.00203	28.5	0.930	0.00794
10	0.00205	28.4	0.929	0.00795

and slightly worse also for length 4. For longer sequences, it was stable. For input length 10, the performance even slightly decreased. A possible reason could be that the network was trained on sequences with length 10 and, therefore, it was not directly trained to predict the 11th frame after inputting 10 frames. To investigate the influence of sequence length during training, we trained a network with $l_s = 5$—half of the basis sequence length—and evaluated it on the 6th frame after inputting 5 frames. We performed two repetitions with 1000 epochs and two repetitions with 2000 epochs. The results in all four cases were similar; SSIM was 0.924 in all cases, PSNR varied from 28.2 to 28.4 and MSE was between 0.00209 and 0.00215. Therefore, the shorter sequence length during training degraded the performance.

As the training on **BDD100K dataset** required long training (large dataset), we performed only two training repetitions. The performance is evaluated in Table V[8]. Usage of larger dataset led to significant performance improvement in all three

[8]Performance of the network from the other training repetition is: MSE 0.00169, PSNR 29.3, SSIM 0.938.

TABLE V
COMPARISON OF PRECNET PERFORMANCE ON CALTECH PEDESTRIAN DATASET AFTER TRAINING ON KITTI (SAME AS IN TABLE III) AND BDD100K DATASET (SEE SECTION IV-C1 FOR DETAILS).

	Caltech Pedestrian Dataset				
Training Set	#frames	#epochs	MSE	PSNR	SSIM
BDD100K	2M	10000	**0.00167**	**29.4**	**0.938**
BDD100K	41K	1000	0.00201	28.6	0.926
KITTI	41K	1000	0.00205	28.4	0.929

measures. Comparing PreCNet trained on large BDD100K subset (2M) with the models trained on KITTI dataset (see Table III), our model was second in SSIM and third in PSNR and MSE.

In order to evaluate effect of different properties of BDD100K and KITTI datasets on performance, we created a small version of the BDD100K dataset with only approx. 41K frames (similar size as the size of KITTI) and used the same training parameters which were used for training on KITTI. The performance on this dataset was similar to performance on KITTI[9]. This suggests that the "quality" of the training set (BDD100K vs. KITTI) is not the key factor for obtaining better performance in this case. We studied the effect of the number of training epochs as well. Validation loss on the small subset of BDD100K (41K frames) started to increase during training (1K epochs), indicating overfitting. Thus, we can exclude the possibility that training for 10K epochs would further improve performance. Hence, we claim that it is really the dataset size that is the enabling factor for performance and that permitted the results obtained for BDD100K (2M frames, 10K epochs).

5) Qualitative analysis: In Fig. 5, there is a qualitative comparison of PreCNet with other state-of-the-art methods trained on KITTI dataset (see Table III). The way of obtaining the predicted frames used for the analysis is the same as for Quantitative analysis (see the predicted frame at $t = 11$ in Fig. 4). For a qualitative comparison of PreCNet with the model by Jin et al. [40] see Fig. 9 (the predicted frame at $t = 11$).

To assess which of the methods is best through visual inspection is not straightforward; none of the models is better than the others in all aspects and shown frames (excluding PredNet which produced significantly worse predictions). For example, in the last row of Fig. 5, DPG has generally the sharpest prediction but PreCNet predicted the street lamp significantly better. To compare our model with the IPRNN, which significantly outperformed all models in all metrics used, we used the sequences from the IPRNN article [33]. On these sequences (see Fig. 5, first two rows), there is no apparent qualitative difference between the predictions by IPRNN and PreCNet. For example, "STOP" sign in the sequence from the first row is predicted sharper by PreCNet than by IPRNN and the other models.

[9]We performed 3 training repetitions on BDD100K with 41K frames. In Table V, there is performance of the best one (according to SSIM). Performance of the other two is MSE {0.00199; 0.00202}, SSIM {0.925; 0.926}, PSNR {28.6; 28.6}.

In Fig. 6, KITTI and BDD100K (both 2M and 41K) training variants (see Table V) are compared. Usage of large BDD100K dataset (with approx. 2M frames) for training led to significant improvement of all the measures (see Table V) in comparison with training on KITTI dataset. It manifested also in the visual quality of prediction of fast moving cars as you can see in the second and third columns of the figure. The phantom parts of the predicted cars were reduced. It also led to better shapes of the predicted cars as you can see in the prediction in the first column (focus on the front part of the van). On the other hand, in some cases training on BDD100K dataset led to blurrier predictions than training on KITTI (see the last column).

D. Multiple frame prediction

For multiple frame prediction, we used the same trained models which we used for next frame video prediction (see Section IV-C). The network had access to the first 10 frames—same as in next frame video prediction. Then, in each timestep, the network produced next frame and this next frame was used as the actual input (as illustrated in Fig. 7). Therefore, the prediction error between the prediction and input frame was zero.

We briefly explored fine-tuning of the network for multiple frame prediction [10]. To generate multiple future frames, the predicted frames—produced by the pre-trained network for next frame prediction—were used as inputs after inputting 10 frames. The network was trained to minimize the mean absolute error between the predicted and ground-truth frames. According to our preliminary results, this did not significantly improve multiple frame prediction performance.

Please note the different meaning of timestep labels t and T: small t starts at the beginning of a sequence, in contrast with capital T, which starts at the beginning of a predicted sequence (see the timestep labels in Fig. 7). Code needed for generation of the results presented is publicly available [11].

1) Quantitative results: In Table VI, there is a quantitative comparison of PreCNet, PredNet, CrevNet and RC-GAN for multiple frame prediction. The methods obtained sequences with a fixed length (10 for PredNet, CrevNet and PreCNet; 4 for RC-GAN; see Section IV-C4 for explanation) of Caltech Pedestrian Dataset and outputted predictions 15 steps ahead (CrevNet only 12). CrevNet, RC-GAN, PredNet and PreCNet (KITTI) were trained on KITTI. PreCNet was also trained on a subset of BDD100K with size 2M. Values for PredNet and RC-GAN were copied from [37]. Values for CrevNet were taken from [32]. Some values for $T = 1$ from Tables III and V are slightly different because the test set used there was split into non-overlapping sequences with different length (11 vs. 25).

For SSIM, PreCNet trained on KITTI outperformed PredNet until timestep $T = 9$ ($t = 19$) when the values became equal and then PreCNet started to lose. For PSNR, PreCNet started to lose earlier ($T = 6$). RC-GAN and CrevNet outperformed PreCNet in nearly all timesteps for SSIM[10] and RC-GAN also in all timesteps for PSNR.

[10]In $T = 1$, SSIM for PreCNet was 0.930 and for CrevNet 0.925.

Fig. 5. **Qualitative comparison of PreCNet with others state-of-the-art methods on Caltech Pedestrian Dataset.** All models were trained on KITTI dataset. Ten input frames were given (see frames for $t = 8, t = 10$), the next one ($t = 11$) was predicted (RC-GAN used only four input frames – see Section IV-C4 for explanation) by the models (for references see Table III). The images of predictions of other models are copied from original or other cited papers (see references in Table III). Position of the sequences in Caltech Pedestrian Dataset by rows; set06-v013, set06-v000, set07-v011, set10-v010, set10-v009, set10-v010, set06-v009.

TABLE VI
A QUANTITATIVE COMPARISON OF SELECTED METHODS FOR MULTIPLE FRAME PREDICTION. SEE TEXT FOR A DESCRIPTION.

Method		T=1	3	6	9	12	15
PredNet [10]	PSNR	27.6	21.7	20.3	19.1	18.3	17.5
	SSIM	0.90	0.72	0.66	0.61	0.58	0.54
RC-GAN [37]	PSNR	29.2	25.9	22.3	20.5	19.3	18.4
	SSIM	0.91	0.83	0.73	0.67	0.63	0.60
CrevNet [32]	SSIM	0.93	0.84	0.76	0.70	0.65	-
PreCNet (KITTI)	PSNR	28.5	23.4	20.2	18.4	17.2	16.3
	SSIM	0.93	0.82	0.69	0.61	0.56	0.53
PreCNet (BDD100K 2M)	PSNR	29.5	24.6	21.4	19.4	18.3	17.4
	SSIM	0.94	0.85	0.73	0.65	0.59	0.56

We also added PreCNet trained on the large subset of BDD100K to the comparison. Then PreCNet outperformed PredNet in all timesteps for SSIM and most timesteps for PSNR; in timestep $T = 15$ it reversed. However, CrevNet and RC-GAN still outperformed PreCNet in most timesteps. For SSIM, PreCNet had better results than CrevNet and RC-GAN only for predicted frames in $T \in \{1, 3\}$. For PSNR, RC-GAN was outperformed by PreCNet only for $T = 1$.

In summary, PreCNet started with mostly better predictions than its competitors, however, its performance tended to degrade faster for prediction further ahead.

2) Qualitative analysis: The methods were compared using the sequences used in [37]. Moreover, PreCNet was separately compared to the model by Jin et al. (Fig. 9). Fig. 8 provides one example (for another illustration, see *Supplementary Materials – Multiple frame video prediction sequence*). Predictions by PreCNet appear less blurred than those by PredNet and

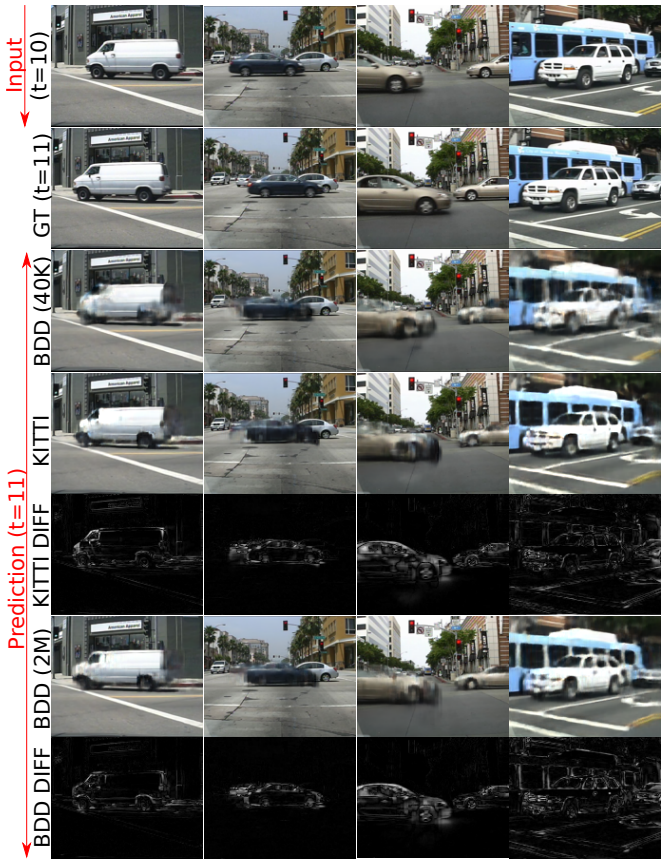

Fig. 6. **Qualitative comparison of PreCNet performance on Caltech Pedestrian Dataset after different training variants.** First row corresponds to the last frame of the input sequence with length 10. Second row corresponds to the ground truth frame. Next rows correspond to the predicted frames of different models which correspond to the models from quantitative evaluation in Table V and related residual images – difference between the predicted and the actual frame. Position of the sequences in Caltech Pedestrian Dataset by columns; set10-v010, set06-v001, set07-v011, set07-v011. In contrast with Fig. 5, the meaning of horizontal and vertical arrangement is inverted. To see whole input sequences and related predictions check *Supplementary materials – Examples of next frame video prediction sequences*.

by the model of Jin et al. (see Fig. 9). This is especially apparent for the later predicted frames. Compared to RC-GAN, predicted frames by PreCNet trained on KITTI seem to have more natural colors and background is mostly less blurred (focus on the buildings in the background). PreCNet trained on large subset of BDD100K (2M of frames) produced even less blurred frames. Comparison with CrevNet is not straightforward. For example, CrevNet captured the geometry of the shadow of the building on the road better than PreCNet. On the other hand, it produced a phantom object (see right side of the road in timesteps $9, 11$) which is not present (or negligible) in the corresponding frames by PreCNet.

V. Conclusion, Discussion, Future Work

In this work, the seminal predictive coding model of Rao and Ballard [2]—here referred to as *predictive coding schema*—has been cast into a modern deep learning framework, while remaining as faithful as possible to the original schema. The similarities and differences are elaborated in detail. We also claim and explain that the network we propose (PreCNet) is more congruent with [2] than others based on the deep learning framework that take inspiration from predictive coding; the case of PredNet [10] is studied explicitly. PreCNet was tested on a widely used next frame video prediction benchmark—KITTI for training (41k images), Caltech Pedestrian Dataset for testing—, which consists of images from an urban environment recorded from a car-mounted camera. On this benchmark, we outperformed most of the state-of-the-art methods and achieved 2nd-3rd rank when measured with the Structural Similarity Index (SSIM)—a performance measure that should best correlate with human perception. Performance on all three measures was further improved when a larger training set (2M images from BDD100k; to our knowledge, biggest dataset ever used in this context) was employed. This may suggest that the current practice based on the rather small KITTI dataset used for training may be limiting in the long run. At the same time, the task itself seems highly relevant, as virtually unlimited amount of data and without any need for labeling is readily available.

Below, we discuss some limitations of this work. For some fast moving objects in the scene, PreCNet could not restore their structure precisely (see e.g., the third column of Fig. 6 where the car contours are not preserved). This may be a drawback of the cost function that minimizes the per-pixel loss. Perceptual loss (e.g., [58]) based on high-level feature differences between frames might alleviate this problem.

In multiple frame video prediction, qualitatively, the frames predicted by PreCNet look reasonable and in some aspects better than some of the competitors. However, a quantitative comparison reveals that PreCNet performance degrades slightly faster than that of its competitors when predicting up to 15 frames ahead. We speculate that architectures which achieve multiple frame prediction by recurrent feeding of previous predictions may not achieve their best performance for next and multiple frame predictions at the same time. Increasing performance for multiple frame prediction may decrease performance for next frame prediction and vice versa. We performed fine-tuning of our network for multiple frame prediction, but preliminary results did not show any significant improvement. PreCNet and predictive coding in general is perhaps intrinsically more suited for next frame prediction. This remains to be further analyzed.

In the future, we plan to analyze the representations formed by the proposed network. It would be interesting to study how much of the semantics of the urban scene has the network "understood" and how that is encoded. For example, our network has not quite figured out that every car has a finite length and its end should be predicted at some point when it is not occluded anymore. In our model, best results on the task were achieved when only prediction error on the bottom level—difference between the actual frame and the predicted one—was minimized during learning. Rao and Ballard [2], on the other hand, minimized this error on every level of the network hierarchy, which may have an impact on the representations formed. Testing on a different task, like human action recognition (e.g., [10], [38], [39]) is also a possibility. Finally, some datasets feature also other signals apart from

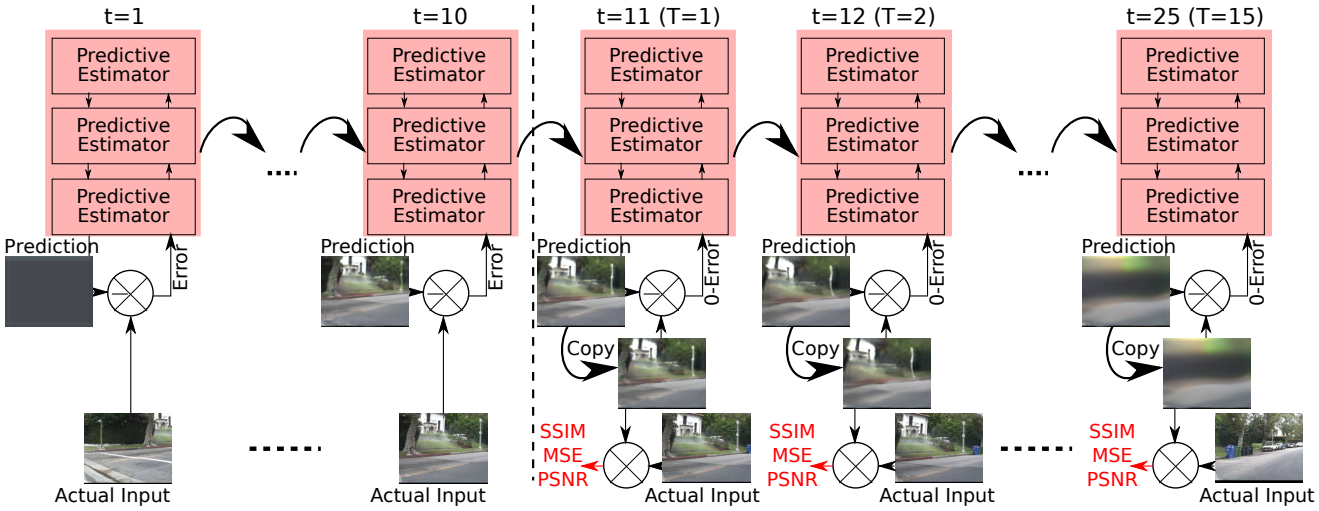

Fig. 7. **Multiple frame video prediction evaluation schema.** After inputting 10 frames, the predicted frames are inputted instead of the actual frames. The prediction errors are therefore zeros. The predicted frames are compared—using MSE, PSNR, SSIM—with the actual inputs. We used this schema for both quantitative and qualitative analysis of Multiple frame prediction (see Section IV-D).

the video stream. Adding inertial sensor signals or the car's steering wheel angle or throttle level is another avenue for future research.

We want to close with a discussion of the implications of our model for neuroscience. Casting the *predictive coding schema* into a deep learning framework has led to exceptional performance on a contemporary task, without being explicitly designed for it. In the future, we plan to analyze the consequences for computational neuroscience. While receptive field properties in sensory cortices remain an active research area (e.g., [59]), a question remains whether the deep learning approach can lead to a better model than, for example, that of Rao and Ballard [2]. Richards et al. [60] and Lindsay [61] provide recent surveys of this perspective. An investigation of this kind has recently been performed for PredNet [23].

Acknowledgment

Z.S. and M.H. were supported by the Czech Science Foundation (GA ČR), project no. 20-24186X. T.S. acknowledges the support of the OP VVV MEYS funded project CZ.02.1.01/0.0/0.0/16_019/0000765 "Research Center for Informatics". The access to the computational infrastructure available through this project is also gratefully acknowledged. We would like to thank to the authors of PredNet [10] for making their source code public which significantly accelerated the development of PreCNet.

References

[1] Y. Huang and R. P. Rao, "Predictive coding," *Wiley Interdisciplinary Reviews: Cognitive Science*, vol. 2, no. 5, pp. 580–593, 2011.

[2] R. P. Rao and D. H. Ballard, "Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects," *Nature neuroscience*, vol. 2, no. 1, p. 79, 1999.

[3] M. W. Spratling, "Reconciling predictive coding and biased competition models of cortical function," *Frontiers in computational neuroscience*, vol. 2, p. 4, 2008.

[4] G. Stefanics, J. Kremláček, and I. Czigler, "Visual mismatch negativity: a predictive coding view," *Frontiers in human neuroscience*, vol. 8, p. 666, 2014.

[5] C. Summerfield and T. Egner, "Expectation (and attention) in visual cognition," *Trends in cognitive sciences*, vol. 13, no. 9, pp. 403–409, 2009.

[6] M. W. Spratling, "Predictive coding as a model of response properties in cortical area v1," *Journal of neuroscience*, vol. 30, no. 9, pp. 3531–3543, 2010.

[7] K. Friston, "Does predictive coding have a future?" *Nature neuroscience*, vol. 21, no. 8, p. 1019, 2018.

[8] A. Clark, "Whatever next? predictive brains, situated agents, and the future of cognitive science," *Behavioral and brain sciences*, vol. 36, no. 3, pp. 181–204, 2013.

[9] M. Mathieu, C. Couprie, and Y. LeCun, "Deep multi-scale video prediction beyond mean square error," in *4th International Conference on Learning Representations, ICLR 2016*, 2016.

[10] W. Lotter, G. Kreiman, and D. Cox, "Deep predictive coding networks for video prediction and unsupervised learning," in *5th International Conference on Learning Representations, ICLR 2017*, 2017.

[11] "Precnet github repository," 2020. [Online]. Available: https://github.com/ZdenekStraka/precnet

[12] M. W. Spratling, "A review of predictive coding algorithms," *Brain and cognition*, vol. 112, pp. 92–97, 2017.

[13] K. Friston, "A theory of cortical responses," *Philosophical transactions of the Royal Society B: Biological sciences*, vol. 360, no. 1456, pp. 815–836, 2005.

[14] Z. Song, J. Zhang, G. Shi, and J. Liu, "Fast inference predictive coding: A novel model for constructing deep neural networks," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 4, pp. 1150–1165, 2018.

[15] M. W. Spratling, "A hierarchical predictive coding model of object recognition in natural images," *Cognitive computation*, vol. 9, no. 2, pp. 151–167, 2017.

[16] K. Han, H. Wen, Y. Zhang, D. Fu, E. Culurciello, and Z. Liu, "Deep predictive coding network with local recurrent processing for object recognition," in *Advances in Neural Information Processing Systems*, 2018, pp. 9201–9213.

[17] H. Wen, K. Han, J. Shi, Y. Zhang, E. Culurciello, and Z. Liu, "Deep predictive coding network for object recognition," in *International Conference on Machine Learning*, 2018, pp. 5266–5275.

[18] S. Dora, C. Pennartz, and S. Bohte, "A deep predictive coding network for inferring hierarchical causes underlying sensory inputs," in *International Conference on Artificial Neural Networks*. Springer, 2018, pp. 457–467.

[19] O. Struckmeier, K. Tiwari, S. Dora, M. J. Pearson, S. M. Bohte, C. Pennartz, and V. Kyrki, "Mupnet: Multi-modal predictive coding network for place recognition by unsupervised learning of joint visuo-tactile latent representations," *arXiv preprint arXiv:1909.07201*, 2019.

[20] A. Ahmadi and J. Tani, "A novel predictive-coding-inspired variational

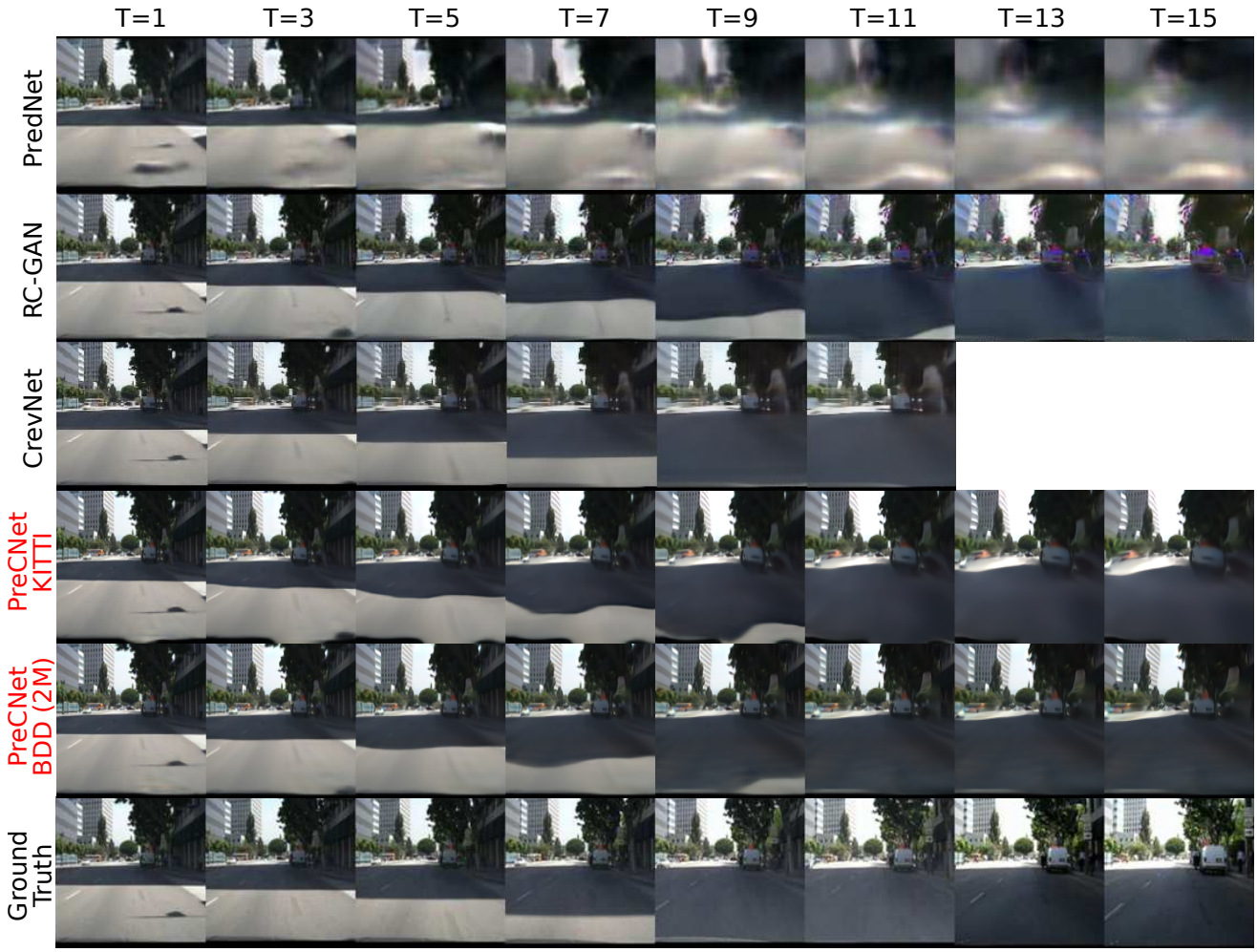

Fig. 8. **A qualitative comparison of selected methods for multiple frame prediction.** The methods obtained sequence with fixed length (10 for PredNet, CrevNet and PreCNet, 4 for RC-GAN; see Section IV-C4 for explanation) of Caltech Pedestrian Dataset and outputted predictions 15 steps ahead. RC-GAN, PredNet, CrevNet and PreCNet (KITTI) were trained on KITTI. PreCNet was also trained on subset of BDD100K with size 2M. This should be noticed during comparison with the other four trained models. This figure was obtained from the figure from [37] by adding sequences for PreCNet and CrevNet (taken from [32]). Location of the sequence in Caltech Pedestrian Dataset is set10-v009. Another qualitative comparison (without CrevNet), with different sequence, is in *Supplementary materials – Multiple frame video prediction sequence*.

rnn model for online prediction and recognition," *Neural computation*, vol. 31, no. 11, pp. 2025–2074, 2019.

[21] M. Choi and J. Tani, "Predictive coding for dynamic visual processing: Development of functional hierarchy in a multiple spatiotemporal scales rnn model," *Neural computation*, vol. 30, no. 1, pp. 237–270, 2018.

[22] R. Chalasani and J. C. Principe, "Deep predictive coding networks," *Proc. Workshop International Conference on Learning Representation*, 2013.

[23] W. Lotter, G. Kreiman, and D. Cox, "A neural network trained for prediction mimics diverse features of biological neurons and perception," *Nature Machine Intelligence*, vol. 2, no. 4, pp. 210–219, 2020.

[24] W. Byeon, Q. Wang, R. Kumar Srivastava, and P. Koumoutsakos, "Contextvp: Fully context-aware video prediction," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 753–769.

[25] F. A. Reda, G. Liu, K. J. Shih, R. Kirby, J. Barker, D. Tarjan, A. Tao, and B. Catanzaro, "Sdc-net: Video prediction using spatially-displaced convolution," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 718–733.

[26] W. Lu, J. Cui, Y. Chang, and L. Zhang, "A video prediction method based on optical flow estimation and pixel generation," *IEEE Access*, vol. 9, pp. 100 395–100 406, 2021.

[27] H. Gao, H. Xu, Q.-Z. Cai, R. Wang, F. Yu, and T. Darrell, "Disentangling propagation and generation for video prediction," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 9006–9015.

[28] Z. Liu, R. A. Yeh, X. Tang, Y. Liu, and A. Agarwala, "Video frame synthesis using deep voxel flow," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4463–4471.

[29] Z. Hao, X. Huang, and S. Belongie, "Controllable video generation with sparse trajectories," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7854–7863.

[30] R. Villegas, J. Yang, Y. Zou, S. Sohn, X. Lin, and H. Lee, "Learning to generate long-term future via hierarchical prediction," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 3560–3569.

[31] C. Finn, I. Goodfellow, and S. Levine, "Unsupervised learning for physical interaction through video prediction," in *Advances in neural information processing systems*, 2016, pp. 64–72.

[32] W. Yu, Y. Lu, S. Easterbrook, and S. Fidler, "Efficient and information-preserving future frame prediction and beyond," in *International Conference on Learning Representations, ICLR 2020*, 2020.

[33] Z. Chang, X. Zhang, S. Wang, S. Ma, and W. Gao, "IPRNN: An information-preserving model for video prediction using spatiotemporal GRUs," in *2021 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2021, pp. 2703–2707.

[34] M. Yuan and Q. Dai, "A novel deep pixel restoration video prediction algorithm integrating attention mechanism," *Applied Intelligence*, vol. 52,

Fig. 9. **A qualitative comparison of PreCNet and model by Jin et al. [40] for multiple frame prediction.** The sequence is from Caltech Pedestrian Dataset and was used in [40]. Both models were trained on KITTI dataset. Input and output sequence lengths were 10. Location of the sequence in Caltech Pedestrian Dataset is set05-v012. As this sequence was used in [40] without downsampling to 10 fps, which is in contrast to other usage of Caltech Pedestrian Dataset in this paper, we left it without downsampling as well.

pp. 5015–5033, 2022.
[35] X. Shu, L. Zhang, G.-J. Qi, W. Liu, and J. Tang, "Spatiotemporal co-attention recurrent neural networks for human-skeleton motion prediction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.
[36] R. Zhang, X. Shu, R. Yan, J. Zhang, and Y. Song, "Skip-attention encoder–decoder framework for human motion prediction," *Multimedia Systems*, vol. 28, no. 2, pp. 413–422, 2022.
[37] Y.-H. Kwon and M.-G. Park, "Predicting future frames using retrospective cycle gan," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1811–1820.
[38] X. Liang, L. Lee, W. Dai, and E. P. Xing, "Dual motion gan for future-flow embedded video prediction," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1744–1752.
[39] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," in *Advances In Neural Information Processing Systems*, 2016, pp. 613–621.
[40] B. Jin, Y. Hu, Q. Tang, J. Niu, Z. Shi, Y. Han, and X. Li, "Exploring spatial-temporal multi-frequency analysis for high-fidelity and temporal-consistency video prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 4554–4563.
[41] A. X. Lee, R. Zhang, F. Ebert, P. Abbeel, C. Finn, and S. Levine, "Stochastic adversarial video prediction," *arXiv preprint arXiv:1804.01523*, 2018.
[42] M. Babaeizadeh, C. Finn, D. Erhan, R. Campbell, and S. Levine, "Stochastic variational video prediction," in *6th International Conference on Learning Representations, ICLR 2018*, 2018.
[43] E. Denton and R. Fergus, "Stochastic video generation with a learned prior," in *International Conference on Machine Learning*, 2018, pp. 1174–1183.
[44] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *Advances in neural information processing systems*, 2015, pp. 802–810.
[45] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[46] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 10, pp. 2222–2232, 2016.
[47] R. P. Rao and D. H. Ballard, "Dynamic model of visual recognition predicts neural response properties in the visual cortex," *Neural computation*, vol. 9, no. 4, pp. 721–763, 1997.
[48] R. P. Rao, "An optimal estimation approach to visual perception and learning," *Vision research*, vol. 39, no. 11, pp. 1963–1989, 1999.
[49] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *International Journal of Robotics Research (IJRR)*, 2013.
[50] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.
[51] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," in *CVPR*, 2009.
[52] ——, "Pedestrian detection: An evaluation of the state of the art," *PAMI*, vol. 34, 2012.
[53] P. Dollár, "Piotr's Computer Vision Matlab Toolbox (PMT)," https://github.com/pdollar/toolbox.
[54] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli *et al.*, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.
[55] Z. Wang, A. C. Bovik, and L. Lu, "Why is image quality assessment so difficult?" in *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4. IEEE, 2002, pp. IV–3313.
[56] S. Winkler, "Perceptual distortion metric for digital color video," in *Human Vision and Electronic Imaging IV*, vol. 3644. International Society for Optics and Photonics, 1999, pp. 175–184.
[57] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations, ICLR 2015*, 2015.
[58] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European conference on computer vision*. Springer, 2016, pp. 694–711.
[59] Y. Singer, Y. Teramoto, B. D. Willmore, J. W. Schnupp, A. J. King, and N. S. Harper, "Sensory cortex is optimized for prediction of future input," *Elife*, vol. 7, p. e31557, 2018.
[60] B. A. Richards, T. P. Lillicrap, P. Beaudoin, Y. Bengio, R. Bogacz, A. Christensen, C. Clopath, R. P. Costa, A. de Berker, S. Ganguli *et al.*, "A deep learning framework for neuroscience," *Nature neuroscience*, vol. 22, no. 11, pp. 1761–1770, 2019.
[61] G. Lindsay, "Convolutional neural networks as a model of the visual system: Past, present, and future," *Journal of Cognitive Neuroscience*, pp. 1–15, 2020.

Zdenek Straka received the BS degree (summa cum laude) in Cybernetics & Robotics and the MS degree (summa cum laude) in Artificial Intelligence from the Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic, in 2014 and 2016, respectively, where he is currently pursuing the Ph.D. degree with the Humanoid and Cognitive Robotics group. His current research interests include neural networks, neurorobotics, computational neuroscience and peripersonal space representations. He received the ENNS Best Paper Award at ICANN 2017.

Tomáš Svoboda received the Ph.D. degree in Artificial Intelligence and Biocybernetics from the Czech Technical University in Prague, Czech Republic, in 2000. He spent three postdoctoral years with the Computer Vision Group, ETH Zurich, Switzerland. He is a Full Professor and Chair of the Department of Cybernetics, FEE, CTU, the Director of Cybernetics and Robotics PhD study program, and he is also on the Board of the Open Informatics programme. He has published articles on multi-camera systems, omnidirectional cameras, image-based retrieval, learnable detection methods, and USAR robotics, he led CTU-CRAS-NORLAB team within DARPA SubT Challenge. His research interests include multimodal perception for autonomous systems, machine learning for better simulation and robot control, and related applications in the automotive industry and robotics.

Matej Hoffmann completed the PhD degree and then served as Senior Research Associate at the Artificial Intelligence Laboratory, University of Zurich, Switzerland (Prof. Rolf Pfeifer, 2006–2013). In 2013 he joined the iCub Facility of the Italian Institute of Technology (Prof. Giorgio Metta), supported by a Marie Curie Intra-European Fellowship. In 2017, he joined the Department of Cybernetics, FEE, CTU in Prague, where he is currently serving as Associate Professor and Coordinator of the Humanoid and Cognitive Robotics group. His research interests are in humanoid, cognitive developmental, and collaborative robotics.