Faculty of Electrical Engineering
Department of Computer Science

Doctoral Thesis

Learning Traversability from Mobile Robot Experience

Miloš Prágr

Copyright

Acknowledgement

Abstract

Understanding mobile robots' traversability over terrains is a crucial component of outdoor autonomous systems, since knowledge of their traversability helps robots to plan safe, fast, and energy-efficient paths. In deployments such as agriculture, forestry, or environment monitoring, a mobile robot can encounter terrains with a priori unknown traversability. The visual appearance and geometry of these terrains might be misleading, such as tall grass that appears as a rigid obstacle when only geometry is considered. The Thesis addresses these challenges by designing a self-improving traversability assessment system. The designed system follows the near-to-far paradigm, where the robot's prior traversal experience is extended to untraversed terrains based on similarities in visual appearance and geometry. The Thesis is presented as a collection of four core publications that address three identified research challenges. The first challenge is focused on learning the traversal experience in a self-improving system, and represents a building block to solve the following challenges. The second challenge focuses on active traversability learning in mobile robot exploration, where the self-improving nature is realized by online decision-making concerning both where to learn the traversability and where to explore the spatial model. The third challenge extends the notion of traversability and thus the scope of the self-improving system through the description of the force to pass through the non-rigid obstacles.

Keywords: Mobile Robotics, Terrain, Traversability, Active Learning, Exploration.

Abstrakt

V nasazení autonomních mobilních robotů je kritické porozumět průchodnosti robotu terénem, kteréžto pomáhá robotu plánovat bezpečně, rychle, a energeticky nenáročné cesty. Při aplikačním nasazení v zemědělství, lesnictví, nebo monitorování prostředí může robot nalézt terény, jejichž průchodnost není předem známá. Vizuální a geometrická podoba takových terénů může být navíc matoucí, jak lze pozorovat na příkladu vysoké trávy, jejíž známá geometrie nasvědčuje tomu, že takový terén je neprůchozí překážkou. V této Práci se navrhuje nasadit v takových prostředích samo se zlepšující systém pro ohodnocování průchodnosti terénem. Navržené systémy přísluší k blízko-dalekým metodám, které rozšiřují předchozí zkušenost robotu s průchodem terénem na zatím neprojité, vizuálně podobné terény. Práce je prezentována jako kolekce čtyř jádrových prací, které řeší tři identifikované výzkumné výzvy. První výzva je zaměřena na učení v samo se zlepšujícím systému ze zkušeností robotu s průchodem terénem, a prezentuje bloky k řešení následujích části Práce. Druhá výzva je zaměřena na aktivní učení průchodnosti v robotickém průzkumu, kde se samo se zlepšují vlastnost systému realizuje v online rozhodování kde se učit průchodnost a kde prohledávat prostředí. Třetí výzva rozšiřuje koncept průchodnosti na popis síly nutné k průchodem poddajnými překážkami.

Klíčová slova: Mobilní robotika, Terén, Průchodnost, Aktivní učení, Průzkum.

Contents

Abbreviations and Acronyms

BCM *Bayesian Committee Machine.*

BCM-GP *Bayesian Committee Machine with Gaussian Process regressor experts.*

BEV *Bird's Eye View.*

CNN *Convolutional Neural Network.*

CoT *Cost of Transport.*

CVaR *Conditional Value at Risk.*

GMM *Gaussian Mixture Model.*

GNG *Growing Neural Gas.*

GP *Gaussian Process.*

GTSP *Generalized Traveling Salesman Problem.*

IGMN *Incremental Gaussian Mixture Network.*

IGNG *Incremental Growing Neural Gas.*

IMU *Inertial Measurement Unit.*

IRL *Inverse Reinforcement Learning.*

ISOINN *Improved Self-organizing Incremental Neural Network.*

LIDAR *Light Detection and Ranging.*

LWPR *Locally-weighted Projection Regression.*

RBCM *Robust Bayesian Committee Machine.*

SLAM *Simultaneous Localization and Mapping.*

SOM *Self-organizing Map.*

SVR *Support Vector Regression.*

TSP *Traveling Salesman Problem.*

UAV *Unmanned Aerial Vehicle.*

VFDT *Very Fast Decision Tree Learner.*

w.r.t. *with respect to.*

Chapter 1

Introduction

The recent advancements in autonomous operation on indoor, urban, and onroad tasks such as autonomous delivery [1] and driving [2] provide a strong foundation for autonomous operation in more complex, offroad environments. During the 2019–2021 period, the research of off-road autonomy was driven by the DARPA Subterranean Challenge [3], a competition focused on search in underground structures, tunnels, and caves. The addressed underground environments are characterized by hard-to-traverse terrains, deteriorated sensing, and operator access being limited by signal attenuation. However, underground autonomy lacks the additional complexity induced by the vegetation that is prevalent on the surface. Hence, the further generation of mobile robotics challenges lies in autonomous agriculture, forestry, and remote environment monitoring. These tasks remain largely unsolved since they require long-term operation in complex, outdoor environments with limited or no human oversight. In particular, complex vegetated environments make it crucial to achieve a high degree of robustness in the robot's autonomous decision-making. Within the decision-making, the assessment of the robot's ability to traverse the environment, denoted as traversability, is vital since understanding the robots' traversability over terrains enhances the decision-making, guiding the robots over safe routes that permit high velocity and low energetic expenditure.

Mobile robots should assess the traversability of each terrain before coming into contact with it, otherwise the robot risks invalidating its plan during the execution. In complex environments such as the forests or caves in Figures 1.1a and 1.1b, an expert-defined traversability model might omit some terra-mechanical properties that are not obvious given the robot's sensor modalities. Even high-fidelity foothold planners such as the local planner used by Belter [4] assume properties such as terrain rigidity or friction. However, it is difficult to ascertain whether a terrain is rigid and thus whether it can support the robot using only the terrain's visual appearance and geometry observable from a distance. Near-to-far models [5], [6] address these challenges by approximating the traversability as a black-box function of the terrain's appearance and geometry learned based on the robot's prior experience of traversing similar-appearing terrains prevalent in the environment. Hence, near-to-far models extend the prior traversal experience to untraversed terrains that look similar to those areas already traversed by the robot.

The research presented in the Thesis[1] aims to build near-to-far traversability predictors suitable for outdoor deployments in agriculture, forestry, or remote inspection. The Thesis is presented as a collection of four core publications, which contribute to the following three research challenges.

---

[1]Further on, this work is referred to as the Thesis, which is authored by the Author.

<div align="center">(a) (b)</div>

<div align="center">(c)</div>

<div align="center">(d)</div>

<div align="center">(e)</div>
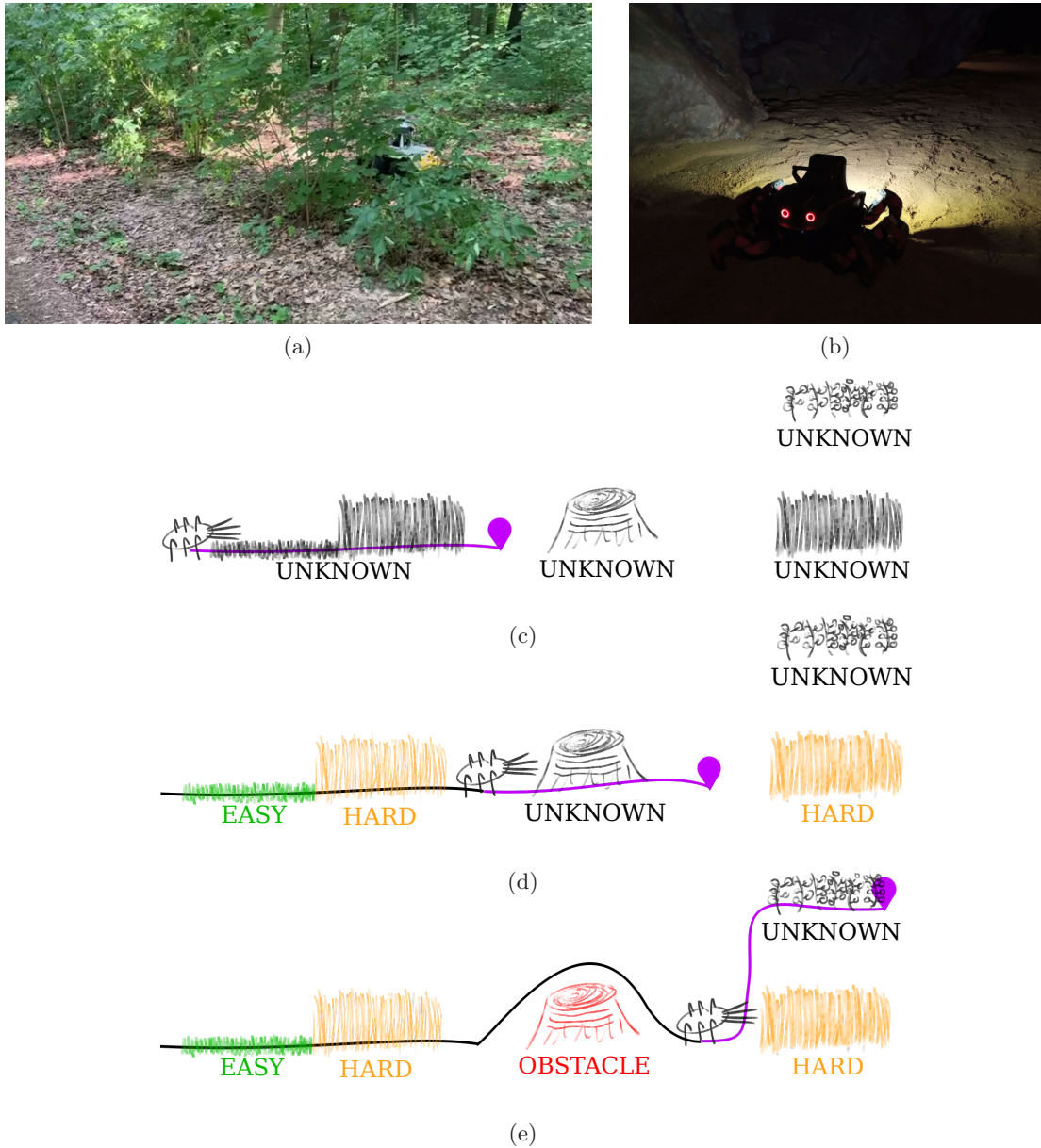
Figure 1.1: (a–b) Considered challenging environments for traversability assessment. (a) Vegetation with different rigidity and through-traversability, depending on the time of the year and other conditions. (b) Cave with deteriorated sensing and operator access limited by signal attenuation. Originally presented in [R7]. (c–e) The self-improving traversability assessment system with active learning. (c) A priori the deployment, the robot has no information about the terrains (unknown terrains in black) and plans to traverse the short and tall grass (goal and planned path in purple). (d) After traversing the short and tall grass (traversed path in black), the robot has learned their respective traversabilities (easy in green, hard in orange) and extrapolates the experience to similar appearing terrains. Next, it plans to traverse over the tree stump. (e) The robot has learned that it cannot traverse over the tree stump, and thus it has avoided the identified obstacle (in red). Next, the robot can traverse one of two terrains: tall grass with known traversability or unknown bush. Since the robot learns the traversability actively, its next goal is to traverse over the bush terrain and thus experience and learn its traversability.

<div align="center">2</div>
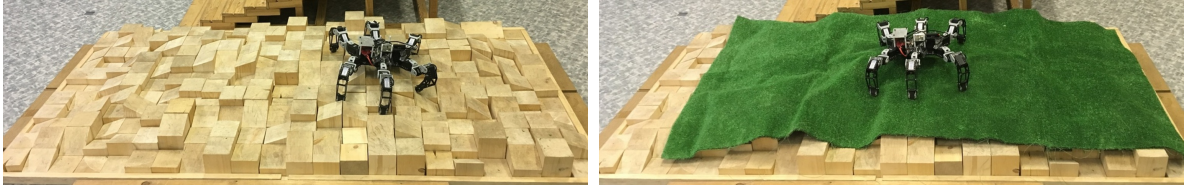
Figure 1.2: A hexapod walking robot traversing rough terrains. The robot pairs the experienced traversal cost to the respective terrain appearance to update its traversability model. Images originally presented in [C1]. © 2018 IEEE.

- **Self-improving Terrain Traversability Assessment.** Missions in outdoor agriculture, forestry, or ground surveillance require long-term autonomous operations in environments where the robot can encounter terrains that are a priori unknown. Therefore, the robot should learn the near-to-far traversability model incrementally from the traversal experience accrued during the deployment. The first research challenge is to design such self-improving terrain traversability predictors, which can adapt or learn from scratch in a new environment, as demonstrated in Figures 1.1c and 1.1d. An example of such a model is presented by Sofman *et al.* [5], who propose an approach for self-supervised online learning of a model inferring *Light Detection and Ranging* (LIDAR)-observed traversability from overhead imagery.

  Hence, the first core publication[2] [C1] builds a near-to-far traversability predictor with different learning and inference viewpoint, which is motivated by traversal assessment over *Unmanned Aerial Vehicle* (UAV)-observed terrains. The scenario is simulated on a laboratory test track traversed by the hexapod walking robot shown in Figure 1.2. The paper investigates the suitability of a set of approaches for terrain description and learning within the near-to-far context, focusing both on incremental and non-incremental scenarios. Besides, [C1] presents a building block for a set of Author's non-core publications that address the following aspects of self-improving traversability assessment, such as the deployment of traversability models in path planning [R1, R2], comparison of incremental learning algorithms [R3, R4, R5], transfer of traversal experience between walking patterns of a hexapod walking robot [R6, R7, R8, R9], influence of the executed motion [R10], and time-series representation of traversability [R11].

- **Online Traversability Learning in Mobile Robot Exploration.** Next, the self-improving nature of the traversability assessment is extended within mobile robot exploration. Mobile robot exploration is an active perception task where one or multiple robots build a map of a priori unknown environment, which is addressed with the additional complexity that the traversability model of the local terrains is also a priori unknown. The novelty is that traversability learning is approached as an active perception problem integrated into the exploration mission. The robot actively creates both the spatial map and traversability model guided by the approximated information gained by interacting with the environment at different sites where new areas can be observed or novel traversal experience can be gained, as illustrated in Figure 1.1e. A similar problem is addressed by Karolj *et al.* [7], who combine spatial exploration with active learning of an underlying spatial magnetism model. The main difference from the addressed problem is that the magnetism model learned in [7] is a function of position,

---

[2]The core publications of the Thesis are marked as [C#], while the Author's non-core publications related to the Thesis are marked as [R#].

3

Figure 1.3: Experimental deployments of online traversability learning in mobile robot exploration. As part of the exploration mission, the robot is tasked to learn its traversability over the terrains in the environment. Images originally presented in (a) [C2] and (b) [C3].

while the near-to-far traversability model is a function of terrain appearance. Hence, the robot can learn the traversability over a particular terrain type at multiple sites in the environment, with the additional complexity of the traversability model affecting the robot's path planning.

The challenge is addressed in two core publications that combine mobile robot exploration and active online traversability modeling into a singular robotic system deployed to learn the traversability of mobile walking robots shown in Figure 1.3. In [C2], the traversability learning and mapping are combined in a myopic manner. Then, the scenario is extended in [C3], where a non-myopic system that can learn multiple traversability models is presented.
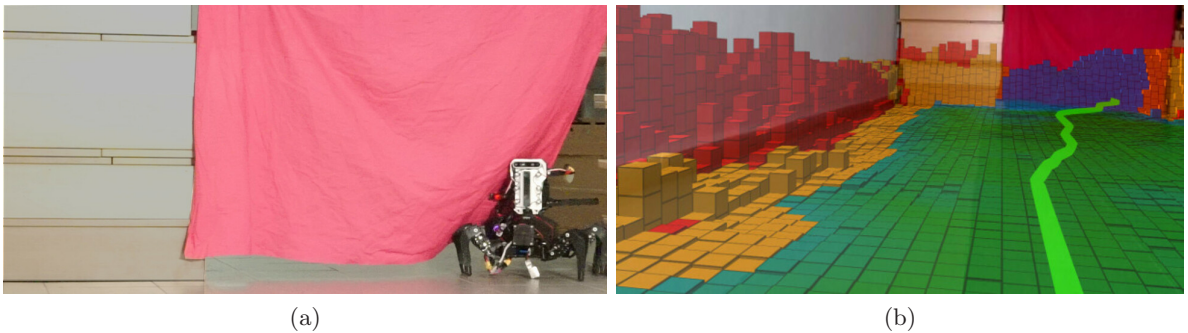
Figure 1.4: Demonstration of the force-to-pass-through model for traversability assessment in an indoor environment. (a) A robot walking through a traversable obstacle in the form of purple fabric. (b) Traversability assessment projected on robot vision. The purple fabric, which is predicted as traversable and thus a valid area for planning the robot's path, is overlayed in blue geometry visualization, while non-traversable areas are in red and orange, and traversable flat ground is in green. Images originally presented in [C4].

- **Traversability Through Non-rigid Terrains.** The third research challenge builds on the self-improving traversability assessment addressed in the previous challenges by focusing on a particular class of terrains such as tall grass that appears as a rigid obstacle

4

based on its geometry. However, such a terrain may traversed through if enough force is exerted by the robot. In the literature, potentially non-rigid terrains are addressed using a haptic antenna [8] or end-to-end learning [9].

In the fourth core publication [C4] of the Thesis, it is proposed to learn a model of the force needed to pass through terrains. The model's predictions are used to determine the traversability through the potentially traversable terrain based on their visual similarity to the terrains the robot has previously attempted to traverse, as demonstrated in Figure 1.4. Compared to the state of the art, the novelty of the proposed solution is in exploiting the effect of the particular terrain on the traversal experience by modeling the terra-mechanical property, which is robot-agnostic. Thus, while the force model is learned from a particular robot's experience interacting with the obstacle using a force sensor, the learned model can be used to determine the traversability of an arbitrary robot assuming the robot's ability to exert forward force is known. Moreover, the proposed approach is deployed in an exploration scenario where it is learned online from a few samples only, compared to the large dataset required by the state-of-the-art end-to-end approach [9].

The structure of the Thesis is as follows. First, a brief overview of the state-of-the-art traversability assessment is provided in Chapter 2. Then, the three research challenges and the Author's related publications are described in Chapters 3 to 5, respectively. Chapter 6 concludes the Thesis and outlines the continuation of the presented work. Appendix A lists the author's publication with citations extracted from the Web of Science. The full texts of the four core publications are presented in Appendices B.1 to B.4.

Chapter 2

Terrain Traversability Assessment

The topic of the Thesis is learning mobile robot traversability from experience, with a particular focus on building self-improving traversability predictors, online active learning of traversability in mobile robot exploration, and learning traversability through non-rigid terrains. Although the reader is referred to the core articles for the detailed review of the state of the art, the next paragraphs provide a short overview of terrain traversability modeling for the convenience of the reader.

Terrain traversability is essential information for mobile robots that are tasked to execute autonomous missions, since they need to understand their ability to traverse the deployment areas. The analysis of such areas yields two types of information. First, it is crucial to identify which areas can be traversed, and which should be avoided outright. Next, the robot selects the traversable areas that provide the best path to its goal considering the path safety, speed, and cost to traverse. The state of the art concerns both problems, often addressing them concurrently. Papadakis [10] presents a taxonomy of the state of the art that splits the approaches into *exteroceptive*, further divided into *geometry-based* and *appearance-based*[3], *proprioceptive*, based on awareness of self-movement and positions, and hybrid, which combine the former two.

The main advantage of exteroceptive approaches is using the terrain's visual and geometric appearance to assess the traversability over terrains from a distance, either by assigning a semantic class or a cost value. Hence, exteroceptive methods are particularly suited to planning, where a decision is needed before traversing the terrain. Moreover, geometric indicators such as terrain roughness, slope, or step height benefit from straightforward interpretation and thus are used to directly assess the traversability of an observed terrain. However, in the literature, the definition of such descriptors is not unified, as demonstrated by the differences in roughness computation reported in [11]–[13]. Besides, different feature types are combined together to improve the terrain descriptive ability, including roughness and Eigen-statistics of point clouds [14], [15], or elevation features and hue-saturation histograms [16]. Visual appearance is particularly relevant when dealing with vegetation with uncertain rigidity, which might be difficult to assess only from geometry, or in settings limited to overhead imagery where the geometric information is not available or very coarse. In vegetation assessment, semantic segmentation over camera images benefits from publicly available off-road datasets RUGD [17] and RELLIS-3D [18]. *Bird's Eye View* (BEV) approaches exploit that top-down view projection of geometric or visual data is well-applicable to *Convolutional Neural Network* (CNN) models such as the semantic traversability segmentation proposed in [19]. Outside of the human-visible spectrum, thermal data [20] and LIDAR sensor reflectance or permeabil-

---

[3]In the remainder of the Thesis, the term *visual appearance* is preferred, since terrain geometry can be considered a component of terrains' appearance, which could invite confusion.

ity [21], [22] assess slippery terrains and vegetation, respectively.

Unlike the visual and geometric appearance, which are based on the robot's exteroception and observed from afar, the robot-experienced difficulty of traversing a terrain is primarily captured by the robot's proprioception, which is its awareness of self-movement and position. Proprioceptive measures are based on robot velocity, energy consumption, vibration, or stability. *Cost of Transport* (CoT) [23] is a measure of energy efficiency of motion originating in biology [24], defined as power consumption over velocity. For wheeled robots, the ride quality can be measured in terms of body vibration, which has adverse effects on the robot's structural integrity, localization, and map quality. Besides, combined visual-vibration models are used to classify terrain types in [25], [26]. Describing stability is of particular interest for multi-legged robots, which can remain statically stable when standing on at least three legs. *Static Stability Margin* [27] and *Dynamic Stability Margin* [28] describe the stability of a multi-legged robot as the distance between the projection of its center of gravity and the closest point on its support polygon defined by its footholds.

Isolated robot-captured experience cannot be used in path planning, since it is limited to the already traversed areas, while the planner needs to assess the difficulty of previously untraversed terrains from a distance. *Near-to-far* approaches learn to infer a traversability indicator characterized by near-robot observations (including robot proprioception) from the far remote observation of the terrain. For instance, Sofman *et al.* [5] use overhead scans and long-range data to infer traversal costs computed from short-range dense point clouds, while Bekhti [6] proposes to infer vibration from terrain images. Besides, *Inverse Reinforcement Learning* (IRL) is used to learn a traversal cost model from human demonstrations, yielding a cost model inducing behavior close to the demonstrated human policy [2]. A recent example [29] addresses IRL convergence issues by using cost with *Conditional Value at Risk* (CVaR), a metric that computes the mean value over a specified tail of a distribution.

Since the Thesis is motivated by instances where geometry and visual appearance without additional context are not sufficient to determine terrain traversability, the robot's proprioceptive experience is approached as a traversal cost function of the visual appearance and geometry learned separately for each environment. Therefore, the approaches presented in the Thesis subscribe to the near-to-far paradigm, where the robot's traversal experience captured in the particular environment is used to adapt the traversability model.

Chapter 3

Challenge 1: Self-improving Terrain Traversability Assessment

Near-to-far approaches such as [5], [6] model the robot's traversability based on its previous experience traversing similar appearing terrains. The Thesis addresses near-to-far predictors in deployments where terrains with novel traversal experience can be encountered. The first research challenge of the Thesis is to build self-improving near-to-far predictors that learn incrementally when a priori unknown terrains are traversed.

The first core publication [C1] is focused on near-to-far models in settings with different learning and inference viewpoints.

> [C1] M. Prágr *et al.*, "Cost of transport estimation for legged robot based on terrain features inference from aerial scan," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 1745–1750. DOI: 10.1109/IROS.2018.8593374. Full text available in Appendix B.1.

The therein presented work is motivated by large-scale traversability assessment from an UAV, where the traversability model learned using the robot's low-to-ground viewpoint is used to predict traversal cost over terrains observed from a different, top-down viewpoint. The groud-to-aerial setup is created by first guiding a hexapod walking robot over a set of terrains of varying difficulty, geometry, and visual appearance. The data captured by the robot are used to learn the traversability model, which is then used to evaluate the traversed terrains using data captured from a viewpoint elevated above the test track. The primary contribution towards the self-improving traversability assessment is benchmarking the performance of several incremental and non-incremental regressors, namely non-incremental *Support Vector Regression* (SVR) [30], non-incremental regression trees, incremental *Hoeffding trees* [31], which discretize the traversal cost into 10 classes, and the *Incremental Gaussian Mixture Network* (IGMN) [32]. Further, several combinations of visual and geometric terrain descriptors are used to learn the traversability models. A combined terrain color and shape descriptor paired with either the IGMN or regression trees is the best performer in the incremental and non-incremental setups, respectively, suggesting that both visual appearance and geometry are of use for viewpoint-robust terrain evaluation. Within [C1], the contributions of the Author are in benchmarking the features and learning approaches.

The first core publication presents a building block for further contributions to self-improving near-to-far traversability prediction. In the non-core publication [R1], the IGMN predictor identified as the best incremental performer in [C1] is used in a path planning scenario. Figure 3.1 illustrated the changes to the cost efficient path induced by incremental
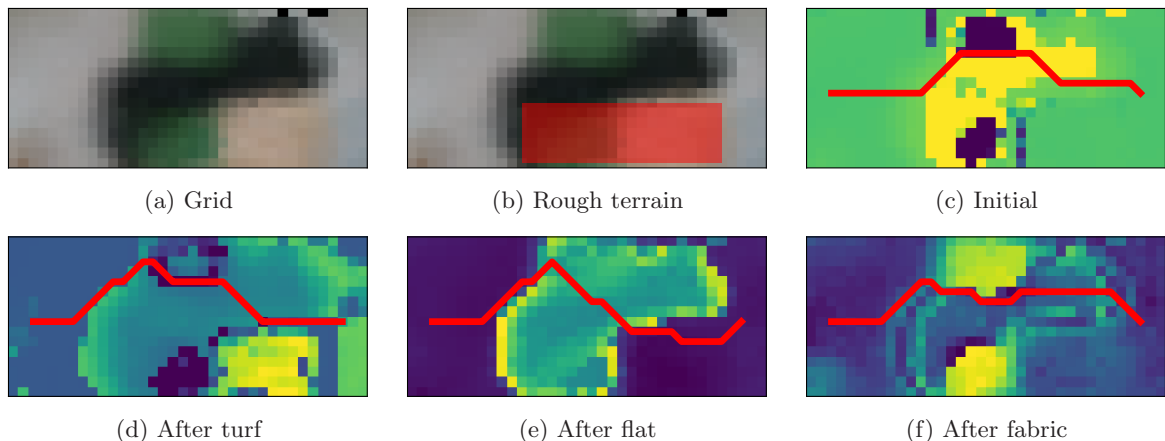
(a) Grid	(b) Rough terrain	(c) Initial
(d) After turf	(e) After flat	(f) After fabric

Figure 3.1: Illustration of the incremental learning of CoT. (a) 0.1^2 m^2 sized squared cell grid over an overhead scan with the (b) rough terrain highlighted and its (c–f) CoT annotations with the planned path over the track. The annotations correspond to the four models sampled (c) at the beginning of the run; after traversing (d) the artificial turf; (e) the flat polyvinyl chloride flooring; (f) and the black fabric. Plots originally presented in [R1].

updates to the traversal cost model. Besides, a multi-goal path planning scenario is addressed in the non-core publication [R2]. *Locally-weighted Projection Regression* (LWPR) [33], an incremental approach popular for its performance in high dimensional spaces, is used to assess traversability from overhead imagery in multi-goal inspection. The sequence of goal visits is modeled as an instance of the *Traveling Salesman Problem* (TSP) where the distance matrix is computed *with respect to* (w.r.t.) the predicted traversal costs.

Since IGMN used in [C1] and [R1] has quadratic time complexity w.r.t. the input dimensionality, the next set of contributions focuses on alternative approaches for traversability learning. The non-core publications [R3] and [R4] investigate using *Self-organizing Map* (SOM) [34] and *Growing Neural Gas* (GNG) [35] for traversability learning, respectively.

The performance of additional incremental learning approaches is investigated in [R5], which extends the benchmarking in [C1] with a focus on incremental learning. The evaluation is computed over robot training data organized as individual trails over the testing track, which are used to learn the individual regressors. The presented benchmarking uses a novel setup where predictions over an observed area larger than the trails are considered since they represent a more realistic path-planning scenario where the robot must infer the cost over yet untraversed areas. Consequently, the robot-experienced costs are not available for the untraversed areas, a setup that prohibits using the experienced costs as the ground truth. Instead, the predictions of a reference non-incremental *Gaussian Process* (GP) regressor [36] are treated as the ground truth for the evaluated incremental predictors. Since the GP regressor infers not only the prediction mean, but also the prediction variance, the incremental regressor prediction is reported as correct if it falls into the 95 % confidence interval of the GP reference model. Moreover, a qualitative evaluation focuses on the suitability of the predicted cost maps for path planning.

Besides the baseline IGMN, the Author tested GNG, *Improved Self-organizing Incremental Neural Network* (ISOINN) [37], LWPR, and *Bayesian Committee Machine* (BCM) [38] with GP regressor experts (BCM-GP). IGMN, ISOINN and LWPR are the best performers, providing both good correctness and cost maps, while GNG suffers from concept drift. BCM-GP performs poorly, contradicting the initial intuition given by its deployment in exploration,
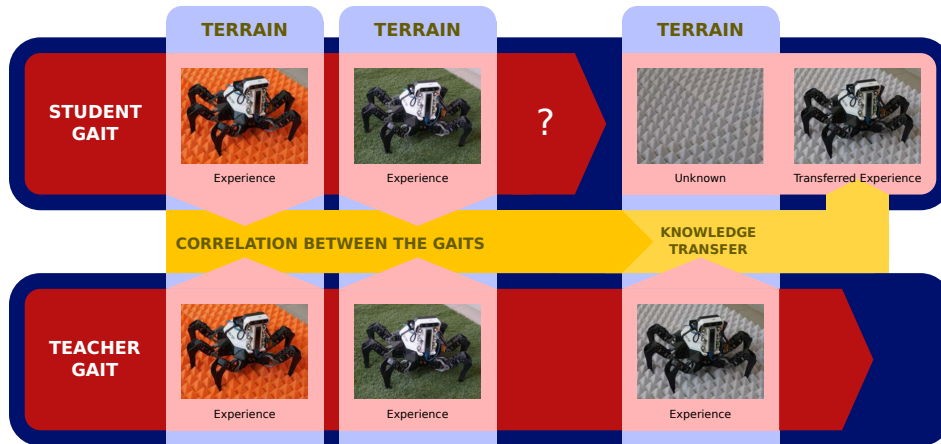
Figure 3.2: The transfer of traversal experience between motion gaits of a hexapod walking robot. The transfer is based on the assumption that if the robot has a similar experience when traversing the green and gray terrains using the first gait (denoted teacher, the source of the traversal experience transfer), but its experience is limited to only one of the terrains (green) with another gait (denoted student, the target of the traversal experience transfer), the experience with the student gait would also match on both the green and gray terrains. Schema originally presented in [R8]. © 2022 IEEE.

which can be attributed to the active perception in exploration where the robot learns until all observed terrains are sufficiently known.

The next set of contributions addresses that the robot's experience is closely tied to the particular motion executed by the robot or its configuration. The non-core publications [R6, R7, R8, R9] are focused on transferring knowledge between individual robots and their configurations. In particular, in [R8], it is proposed to transfer experience between the walking gaits of a multi-legged robot using a system of *Gaussian Mixture Model* (GMM) regressors. The proposed system is designed to infer terrain-gait combinations that were not experienced in learning, as overviewed in Figure 3.2.

In [R10], the feature description of the traversed terrain is extended by a feature description of the motion executed by a hexapod walking robot. The robot's plans are constrained by a simplified mask of the robot's footholds, which is applied to the robot's position at the start and end of the planned motion. The individual constraints are designed to ensure the robot's pose stability, constraining the relative elevation of different footholds in the goal state and the relative elevation of individual footholds between the start and end state. For valid motions, a near-to-far model is learned that infers stability-based traversal cost from a combination of terrain descriptors and novel descriptors of motion. The design of the motion descriptors follows the experimental verification that forward motion is preferable over turning, and that uphill and downhill locomotion might exhibit different robot stability. Hence, the descriptors capture the slope (relative elevation) of the planned path over traversed terrain and indicate turning.

Since the experience-based traversability is computed from a sequence of robot's proprioceptive measurements while the robot gradually moves over the traversed area, at each point of the traversal the robot may experience different terra-mechanical properties related to the terrains under its footprint. Borders between multiple terrain types are of particular interest, since they can exhibit different properties than any of the two terrains altogether. These properties of terrain traversability suggest that the traversability inference mechanism would benefit from modeling the succession of the traversed terrains. In the contributed [R11], it is
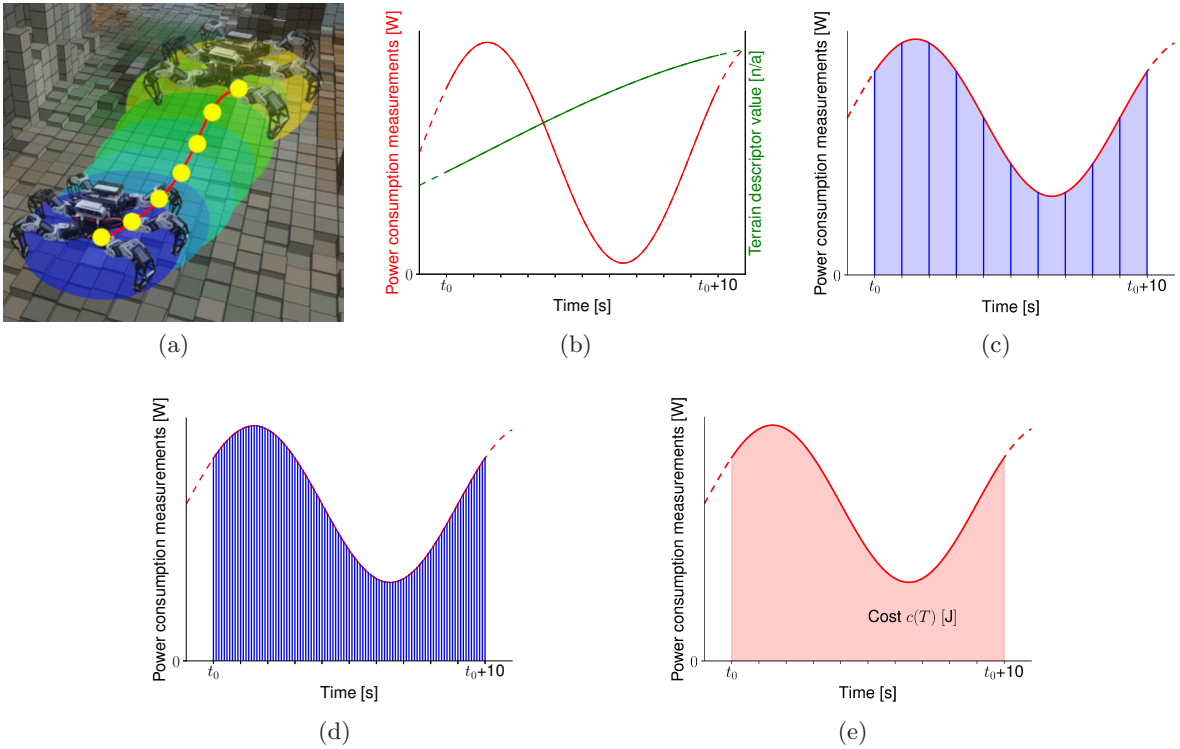
Figure 3.3: Illustration of traversability assessment with time series. (a) Visualization of a hexapod walking robot traversing the modeled 10 s long segment, (b) the associated power consumption and terrain descriptor signals, (c) segment discretization into ten 1 s long intervals visualized over the power consumption signal, (d) all the power consumption measurements captured over the segment, and (e) the ground truth energy cost of the segment computed as an integral over the sampled power consumption values. Plots originally presented in the talk given for [R11].

proposed to model the terrain traversal as a time series. The traversal is split into 10 s long segments, represented as sequences of regularly sampled points carrying the robot's sampled proprioceptive signals and terrain descriptors observed at the respective point locations. Figure 3.3 shows a time series segment and the associated signals. The proprioceptive signals are used to characterize an energy consumption-based cost, which is cumulative with distance and thus suitable for time series modeling. The terrain feature signals are used to predict the segment's traversal cost and to classify the segment points using a GNG model. The time-series models with different point granularity (11 points in 10 s shown in Figure 3.3c) are compared to a non-series baseline, and the time series representing the traversal at 2.5 Hz (26 points in 10 s) perform the best.

Chapter 4

Challenge 2: Online Traversability Learning in Mobile Robot Exploration

The challenge addressed by the two core publications [C2] and [C3] is to actively learn the robot's terrain traversability model as a part of its exploration mission. Mobile robot exploration is an active perception task where one or multiple robots build a spatial map of a priori unknown environment, which helps the robot reason how to navigate the environment. Frontier-based exploration is considered the fundamental approach to spatial exploration [39]. It is based on following frontiers, the borders between the observed traversable areas and the unknown space [40]. Alternatively, the probabilistic representation of the cell occupancy on occupancy grids is used to select an exploration strategy that maximizes the approximate information gain about the grid [41], [42], and Cauchy-Schwarz quadratic mutual information may be employed to compute the information gain efficiently [43]. Besides, robot location uncertainty may be incorporated into information-based exploration approaches as the localization uncertainty represented using the differential entropy of the robot position Gaussian distribution [41], [44], [45].

The addressed problem of active traversability learning is related to modeling phenomena underlying the environment such as the spatial distribution of temperature [46], spread of gas [47], or spatio-temporal models of ocean salinity [48]. In the co-authored non-core publication [R12], the signal strength in subterranean environment is modeled as a function of the relative position w.r.t. the transmitter and a feature descriptor of the environment geometry. Active spatial-based modeling is an example of informative path planning [49], a problem to find the most informative path through the environment [50] subject to constraints such as an energy budget [51]. Informative path planning approaches can be broadly divided into myopic methods, which are greedy and plan only concerning the next goal, and non-myopic methods, which consider longer horizons. For example, in frontier-based exploration, a myopic explorer navigates towards the closest frontier, while a non-myopic explorer plans a path that visits all the frontier representatives [52].

In the state of the art, active learning of traversability is proposed by Mayuku *et al.* [53] who learn to infer vibration traversability from images with self-supervised data gathering, but the therein prosed system is not considered within the exploration or online learning contexts. Outside of traversability modeling, closest to the addressed problem is the method proposed by Karolj *et al.* [7], who combine frontier-based exploration with active learning of an underlying spatial magnetism model. However, while the state of the art regarding the simultaneous map and model exploration is, to the best of the Author's knowledge, limited to spatial-based models, it is desirable to infer traversability from terrain appearance or geometry as presented in the core publications in the following paragraphs.
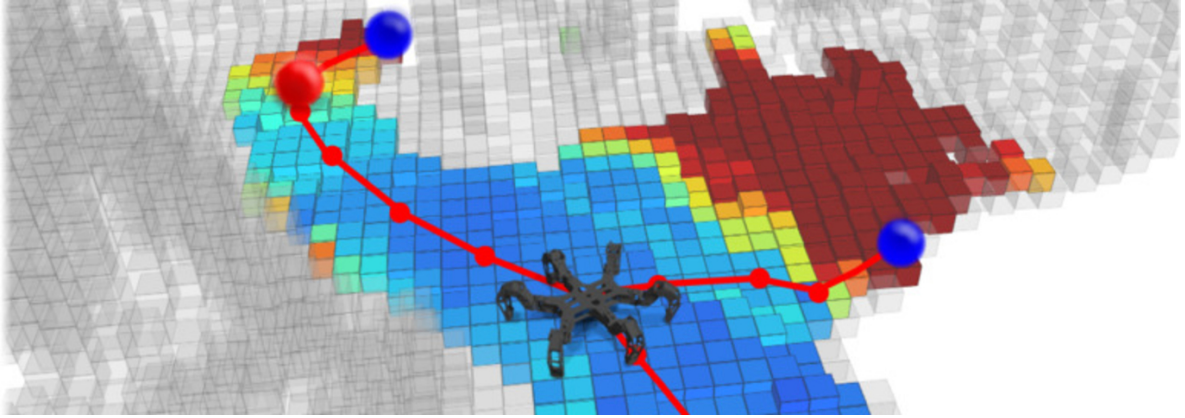
Figure 4.1: A visualization of active traversability learning in mobile robot exploration. The possible learning and exploration goals are depicted over an elevation grid map where easy terrains are in blue, difficult terrains are in red, and untraversable areas are in gray. The robot can navigate either to spatial exploration goals at the borders of the traversable mapped environment and unknown space (blue spheres) or traversability model learning goals located at yet untraversed terrains (red spheres). Given the myopic decision-making employed in the first proposed system, the robot would select the traversability exploration goal. Figure originally presented in [C2].

The first core publication addressing the second challenge [C2] leverages the work on incremental traversability learning in an active learning scenario.

> [C2] M. Prágr *et al.*, "Online incremental learning of the terrain traversal cost in autonomous exploration," in *Robotics: Science and Systems (RSS)*, 2019. DOI: 10.15607/ RSS.2019.XV.040. Full text available in Appendix B.2.

In the proposed system, the robot creates its traversability model online from its experience of terrain traversal captured while pursuing cost model exploration goals as a part of an environment exploration strategy. In the exploration, the robot thus seeks both the spatial exploration goals in the form of frontiers, and cost exploration goals in the form of the sites that are the most informative w.r.t. the traversability model, both of which are visualized in Figure 4.1. The traversability inference and goal identification are done via a GP. However, unlike Luo and Sycara [46] or Karolj *et al.* [7], the model infers the costs based on terrain appearance, not its position in the environment. Besides, since GP regression learning time complexity scales cubically with the number of training samples, the training set is split incrementally into fixed-sized experts that are combined into a product-of-experts using the BCM. The cost model and spatial exploration are combined in a myopic manner, with the robot seeking the cost learning goals first so that it limits planning over terrain with unknown traversability. The contributions of the Author are particularly focused on the terrain feature description, the traversability inference using BCM-GP, the cost-learning goal identification, and the integration of the learning goals into the exploration strategy.

An explorer that aims to use non-myopic decision-making must address how to combine the expected information gained from the different models. A similar issue is addressed in mobile robot exploration with position uncertainty, where the robot aims to remain certain of its position since poorly localized measurements are of limited utility for mapping. In [41] and [44], the total exploration utility is a linear combination of the occupancy uncertainty, represented by Shannon's discrete entropy computed from cell obstacle probabilities, and the robot

localization uncertainty, represented using the differential entropy based on its position distribution. Carrillo *et al.* [45] argue that combining Shannon's discrete and differential entropies is not desirable due to scaling issues, and they propose to use the Rényi entropy [54] of the occupancy grid weighted by the localization uncertainty. The issue of combining discrete and differential entropies is also prevalent for terrain learning in exploration, since the grid-based spatial models yield the discrete entropy, while the traversal cost regressors yield differential entropy. The solution proposed by Carrillo *et al.* [45] is designed for weighting by position uncertainty and thus is not suitable for the traversability learning scenario. In [C2], the issue is avoided by opting for an exploration strategy where one of the models is dominant. In scenarios with several models with no clear hierarchy, the approach proposed in [C2] cannot be used.

The next core publication [C3] addresses the non-myopic scenario with multiple models.

> [C3] M. Prágr *et al.*, "Autonomous robotic exploration with simultaneous environment and traversability models learning," *Frontiers in Robotics and AI*, vol. 9, 2022. DOI: `10.3389/frobt.2022.910113`. Full text available in Appendix B.3.

The proposed system, which is overviewed in Figure 4.2a, considers a path to improve both the spatial and multiple traversal cost models, each corresponding to a particular configuration of a multi-legged robot's gait. The issue of combining the two incompatible measures of uncertainty is avoided by using a policy that links the informative sites separately identified for spatial and cost exploration. During the exploration, the spatial model and each cost model provide a set of goals to visit, representing frontier areas and terrain types with uncertain traversal costs. Since similar terrains may be prevalent at multiple locations, each terrain type goal may be represented by multiple learning sites. For each goal, visiting one of the sites is sufficient, and hence the exploration site sequence is selected as a solution of the open-ended *Generalized Traveling Salesman Problem* (GTSP) [55], a variant of the TSP where nodes (learning sites) are grouped into exclusive and exhaustive sets (goals) and the problem is to visit each set. An example instance of the GTSP for system with one traversability model is shown in Figure 4.2b.

The individual terrain-gait models are based on GP regressors, and the size of their respective learning sets is limited by accruing experience only when unknown terrain is encountered or erroneous prediction is experienced. Each terrain-gait model also comprises an *Incremental Growing Neural Gas* (IGNG) [56] employed to cluster similar-appearing terrains and thus to create the set of learning goals for each terrain-gait model. The contributions of the Author are in building the cost learning modules, terrain type identification and learning goal selection, and in designing the combined exploration policy.
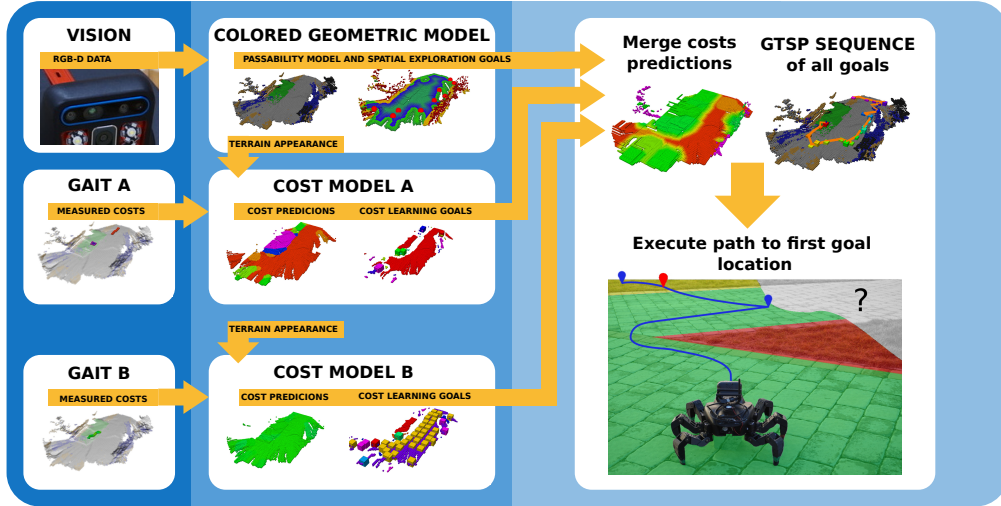
(a)
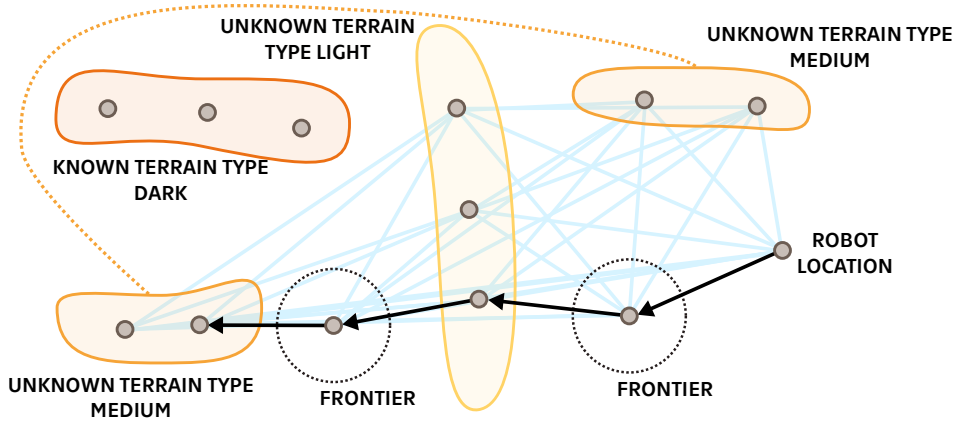
(b)

Figure 4.2: (a) An overview of the non-myopic exploration system with online active traversability model learning. The robot builds an environment color-elevation model using its sensor measurements, which are used to determine areas that can be traversed by the robot. For each motion gait of the hexapod walking robot, the terrain appearance is paired with the respective traversal experience to learn the gait-cost models, which are used to plan the exploration path. During the exploration, the robot navigates to goals independently determined to explore the spatial model and each gait-cost model. The sequence of goals to visit is determined as a solution to an instance of the GTSP. Schema originally presented in [C3]. (b) Illustration of the open-ended GTSP to determine site sequence to sample unknown terrains' traversability and visit frontier locations. In the environment, there are two frontier locations and three terrain types: *dark*, *medium*, and *light*. The *medium* and *light* terrains have unknown traversability, and thus should be visited to learn their respective traversabilities. The traversability of the *dark* terrain is known, and therefore the *dark* terrain does not have to be visited and is excluded from the GTSP instance. The traversability of each terrain can be learned at multiple sampling sites (small circles), and it is sufficient to visit only one site per each terrain. The regions covering a particular terrain can be disconnected, as is the case for the *medium* terrain. Unlike the terrains, each frontier location corresponds to exactly one sampling site. The GTSP problem is to visit exactly one site per set (terrain/frontier), and thus the inter-set edges are considered (edges simplified as Euclidean in the schema, in reality affected by traversal cost). The shortest exploration path (path edges in black, other edges in light blue) hence covers one site for each frontier and unknown terrain type.

Chapter 5

Challenge 3: Traversability Through Non-rigid Terrains

Terrains such as tall grass are not rigid and thus might be traversable given enough force exerted by the robot. However, in a geometric model, such terrains appear as obstacles. While assuming non-rigid objects to be non-traversable leads to safe behavior since potential obstacles are avoided, the resulting decision-making is suboptimal and might prevent the robot from reaching areas separated by such apparent barriers. In the state of the art, a similar problem is addressed by Baleia *et al.* [8], who use a haptic antenna to determine the traversability of potentially traversable obstacles. When the robot encounters an unknown obstacle, it recalls the k-nearest appearing obstacles to decide whether the obstacle can be moved through or avoided. When neither decision can be made with sufficient confidence, the antenna is used to assess the obstacle, further expanding the robot's memory. Besides, potentially traversable terrains are addressed in the self-supervised, end-to-end system presented by Kahn *et al.* [9]. The therein proposed system uses a random walk policy to collect a dataset of collision, bumpiness, and position events using its *Inertial Measurement Unit* (IMU) and wheel odometry. The robot learns to predict the events given input image and action. The learned models can be exploited in navigation through potentially traversable terrains without relying on a *Simultaneous Localization and Mapping* (SLAM) system w.r.t. an arbitrary reward function that considers the collision, bumpiness, and position events.

In the fourth core publication [C4], it is proposed to model the traversability through potentially obstructing obstacles as the force needed to pass through.

> [C4] M. Prágr *et al.*, "Autonomous exploration with online learning of traversable yet visually rigid obstacles," *Autonomous Robots*, vol. 47, pp. 161–180, 2023, Springer. DOI: 10.1007/s10514-022-10075-4. Full text available in Appendix B.4.

Assuming the robot's ability to exert force forwards is known, the proposed force model provides the robot with information that enables navigation through non-rigid obstacles as visualized in Figure 5.1. The force is sampled by a robot using a bumper device, and the force measurements are connected to terrain appearance descriptors to infer the force needed to pass through yet unsampled terrains. Hence, the robot learns whether it can pass through obstacles that appear rigid but might be traversed through given sufficient force. Since the force required to pass through is a property of the terrain and is independent of the robot type, the obstacle traversability learning is robot agnostic. The maximum force that can be exerted in forward motion by each robot is treated as the robot's traversability parameter.

The force to pass through the obstacles is modeled using a GP. The GP is learned incre-
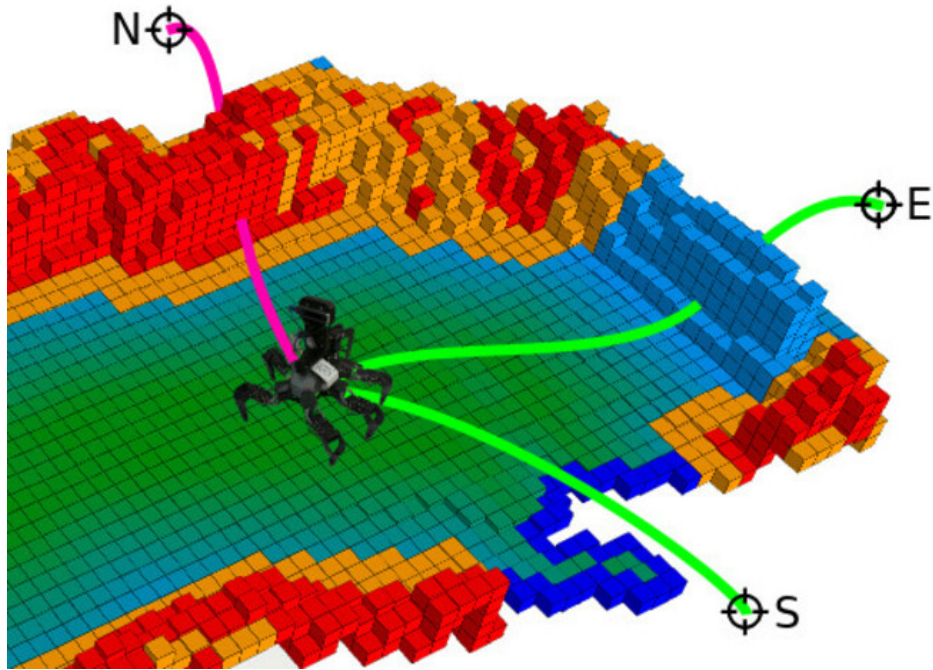
Figure 5.1: A visualization of a robot using the prosed system for assessment of non-rigid obstacles. The southern goal (S) can be reached over flat ground. The northern goal (N) cannot be reached since it is hidden behind a rigid obstacle. The eastern goal (E) can be reached through an obstacle that has been deemed non-rigid and thus traversable based on previous experience with similar terrains. The schema is part of a figure originally presented in [C4].

mentally during the deployment, selecting samples based on the estimated information gained by interacting with the observed terrains in the environment. Unlike in [C2], where the experience is collected continuously, the collected experience is limited to the obstacles and thus relatively sparse. Hence, similarly to [C3], a single GP regressor is used that is continually relearned. The traversability model is deployed both in an outdoor environment with real vegetation, and on a hexapod walking robot in an indoor autonomous escape scenario where the robot is trying to find a path out of an arena enclosed by apparent obstacles, some of which are traversable. In the escape scenario, the robot is set to explore the environment to build a complete traversability map of an unknown environment with areas "hidden" by non-rigid terrains visually appearing as obstacles. There, the system demonstrates online learning and decision-making based on the expected information gained from visiting and interacting with the obstacles. Since the goal of the robot is to escape the constraints of the given area, the obstacle and spatial exploration are combined in a myopic manner. However, unlike in [C2], where traversability goals are preferred over spatial exploration, the robot tries to "escape" the environment by seeking the spatial exploration goals first and resorts to learning the obstacle rigidity only when it recognizes it is enclosed by obstacles with unknown rigidity. The Author's contributions are particularly focused on the force-based traversability model, its incremental learning, the identification of the learning goals, and the integration of active model learning into the exploration scenario.

The main difference between the state-of-the-art [9] and [8], and the core publication [C4] is twofold. First, compared to [9], which is learned using a random-walk data collection, the developed solution requires only few samples to connect a particular terrain appearance with

the force to pass through. Besides, the developed solution aims to model a robot-agnostic terrain property, which is not the case for the end-to-end system proposed in [9]. Second, the solution is designed to operate over a geometric map of the environment and can operate in online learning in mobile robot exploration. Hence, the force model can be integrated into the non-myopic multi-model learning system proposed in [C3].

Chapter 6

Conclusion

The Thesis is motivated by deployments without prior learning in environments where geometric and visual appearance alone is not sufficient to assess the terrain traversability due to ambiguity or changing conditions. There, establishing an inference mechanism that learns to predict the future traversal experience from the appearance is desirable. Hence, the aim of the Thesis is to create a near-to-far system capable of online, active learning of mobile robot traversability. The Thesis is presented as a collection of four core publications focused on three research challenges.

The first challenge concerns building the self-improving terrain traversability assessment model from the mobile robot experience. The target system addresses environments where the robot encounters terrains with a priori unknown traversability, and hence must adapt its traversability assessment model incrementally. The first core publication [C1] concerns near-to-far predictor where the learning and prediction viewpoints differ, and compares a set of geometric and visual appearance terrain descriptors, and non-incremental and incremental learning approaches. [C1] opens a research stream comprising a set of non-core publications concerning the use of incrementally learned traversability in path planning [R1, R2], benchmarking incremental learning in near-to-far traversability assessment [R3, R4, R5], transfer of traversal experience between mobile robots and their configurations [R6, R7, R8, R9], motion characterization in traversal assessment [R10], and time-series-based traversability modeling [R11].

The second challenge extends the self-improving nature of the traversability learning system through active traversability model learning during mobile robot exploration. In the active learning scenarios, the robot solves the decision-making problem where to learn to improve the model. Similarly, in mobile robot exploration, the robot is deployed in an environment that is at least partially unknown and tasked to move around to build a map of its surroundings. Since in exploration, it is likely that the traversability model is also a priori unknown, the robot is tasked to move between different sites to either map new areas or experience the traversal over novel terrains. The challenge comprises two core publications. The second core publication of the Thesis [C2] presents a system for online incremental learning of a terrain traversal cost in autonomous exploration. The explorer seeks both the frontier areas of the spatial model, and the cost learning goals in the form of areas deemed the most informative by the incrementally learned GP cost regressor. The cost model and spatial exploration are combined in a myopic manner, with the robot seeking the cost learning goals first so that it limits planning over terrain with unknown traversability. The third core publication [C3] addresses the myopic nature of the preceding systems. Since it is desirable to avoid combining the measures of informativeness of the cost prediction and spatial models, a policy is used that combines informative sites separately identified for spatial and cost exploration. The

sequence of sites to visit, and thus the informative path, is determined as a solution of the GTSP.

The third challenge addresses that the prior contributed work assumes that observed obstacles are rigid and thus non-traversable. However, some terrains with obstacle-like geometry, such as tall grass, might be actually traversable if enough force is exerted by the robot. In the fourth core publication [C4], it is proposed to learn a model of the force needed to pass through the apparent obstacles. Since the model describes the resistance of the terrain, the resulting traversability model is robot agnostic, assuming that the maximum force that can be exerted by each particular robot is known. The active learning of the proposed model based on estimated information gained by interacting with the obstacles is demonstrated in an exploration scenario similar to [C2] and [C3].

6.1 Possible Extensions

The overviewed publications present the building blocks for self-improving traversability models. The individual publications concern incremental learning, extend the modality of the learned traversability phenomenon w.r.t. motion characterization and non-rigid terrains, or concern traversability learning in path planning and exploration. The individual components provide various areas that can be further refined. For instance, the presented contributions assume that the cost model is learned incrementally from scratch, while in practical scenarios, the model might be partially known and needs only limited adaptation, posing a transfer learning problem. Besides, there is a rich set of possible extensions of modalities of traversability. The contribution regarding non-rigid, potentially traversable obstacles can be followed by addressing non-rigid ground that might prove untraversable although it appears traversable, which is an inverse of the problem addressed in [C4]. Furthermore, both types of non-rigid terrains are likely to change their appearance and terra-mechanics after robot traversal, which might affect multi-robot teams.

Bibliography

◼ Core Publications

[C1] M. Prágr, P. Čížek, and J. Faigl, "Cost of transport estimation for legged robot based on terrain features inference from aerial scan," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 1745–1750. DOI: 10.1109/IROS.2018.8593374.

[C2] M. Prágr, P. Čížek, J. Bayer, and J. Faigl, "Online incremental learning of the terrain traversal cost in autonomous exploration," in *Robotics: Science and Systems (RSS)*, 2019. DOI: 10.15607/RSS.2019.XV.040.

[C3] M. Prágr, J. Bayer, and J. Faigl, "Autonomous robotic exploration with simultaneous environment and traversability models learning," *Frontiers in Robotics and AI*, vol. 9, 2022. DOI: 10.3389/frobt.2022.910113.

[C4] ——, "Autonomous exploration with online learning of traversable yet visually rigid obstacles," *Autonomous Robots*, vol. 47, pp. 161–180, 2023, Springer. DOI: 10.1007/s10514-022-10075-4.

◼ Author's Related Publications

[R1] M. Prágr, P. Čížek, and J. Faigl, "Incremental learning of traversability cost for aerial reconnaissance support to ground units," in *2018 Modeling & Simulation for Autonomous Systems (MESAS)*, Springer, 2019, pp. 412–421. DOI: 10.1007/978-3-030-14984-0_30.

[R2] M. Prágr, P. Váňa, and J. Faigl, "Aerial reconnaissance and ground robot terrain learning in traversal cost assessment," in *2019 Modeling & Simulation for Autonomous Systems (MESAS)*, Springer, 2020, pp. 3–10. DOI: 10.1007/978-3-030-43890-6_1.

[R3] J. Faigl and M. Prágr, "Incremental traversability assessment learning using growing neural gas algorithm," in *Advances in Self-Organizing Maps, Learning Vector Quantization, Clustering and Data Visualization (WSOM 2019)*, Springer, 2020, pp. 166–176. DOI: 10.1007/978-3-030-19642-4_17.

[R4] ——, "On unsupervised learning of traversal cost and terrain types identification using self-organizing maps," in *International Conference on Artificial Neural Networks (ICANN)*, Springer, 2019, pp. 654–668. DOI: 10.1007/978-3-030-30487-4_50.

[R5] M. Prágr and J. Faigl, "Benchmarking incremental regressors in traversal cost assessment," in *International Conference on Artificial Neural Networks (ICANN)*, Springer, 2019, pp. 685–697. DOI: 10.1007/978-3-030-30487-4_52.

[R6] R. Szadkowski, M. Prágr, and J. Faigl, "Transfer of inter-robotic inductive classifier," in *International Conference on Automation, Control and Robots (ICACR)*, IEEE, 2020, pp. 32–36. DOI: 10.1109/ICACR51161.2020.9265509.

[R7] J. Zelinka, M. Prágr, R. Szadkowski, J. Bayer, and J. Faigl, "Traversability transfer learning between robots with different cost assessment policies," in *2021 Modeling & Simulation for Autonomous Systems (MESAS)*, Springer, vol. 13207, 2022, pp. 333–344. DOI: 10.1007/978-3-030-98260-7_21.

[R8] M. Prágr, R. Szadkowski, J. Bayer, J. Zelinka, and J. Faigl, "Terrain traversal cost learning with knowledge transfer between multi-legged walking robot gaits," in *IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2022, pp. 148–153. DOI: 10.1109/ICARSC55462.2022.9784790.

[R9] J. Zelinka, M. Prágr, R. Szadkowski, J. Bayer, and J. Faigl, "Deep transfer learning of traversability assessment for heterogeneous robots," in *Conference Information Technologies - Applications and Theory (ITAT)*, vol. 3226, 2022, pp. 12–20.

[R10] M. Prágr, P. Čížek, and J. Faigl, "Traversal cost modeling based on motion characterization for multi-legged walking robots," in *European Conference on Mobile Robots (ECMR)*, IEEE, 2019, pp. 1–6. DOI: 10.1109/ECMR.2019.8870912.

[R11] M. Prágr and J. Faigl, "Terrain learning using time series of ground unit traversal cost," in *2019 Modeling & Simulation for Autonomous Systems (MESAS)*, Springer, 2020, pp. 97–107. DOI: 10.1007/978-3-030-43890-6_8.

[R12] M. Zoula, M. Prágr, and J. Faigl, "On building communication maps in subterranean environments," in *2020 Modeling & Simulation for Autonomous Systems (MESAS)*, Springer, 2021, pp. 15–28. DOI: 10.1007/978-3-030-70740-8_2.

▪ References

[1] C. Chen, E. Demir, Y. Huang, and R. Qiu, "The adoption of self-driving delivery robots in last mile logistics," *Transportation Research Part E: Logistics and Transportation Review*, vol. 146, p. 102 214, 2021. DOI: doi.org/10.1016/j.tre.2020.102214.

[2] M. Wulfmeier, D. Rao, D. Z. Wang, P. Ondruska, and I. Posner, "Large-scale cost function learning for path planning using deep inverse reinforcement learning," *International Journal of Robotics Research*, vol. 36, no. 10, pp. 1073–1087, 2017. DOI: 10.1177/0278364917722396.

[3] T. H. Chung, V. Orekhov, and A. Maio, "Into the robotic depths: Analysis and insights from the DARPA Subterranean Challenge," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 6, no. 1, annurev–control–062722–100728, 2023. DOI: 10.1146/annurev-control-062722-100728.

[4] D. Belter, "Informed guided rapidly-exploring random trees-connect for path planning of walking robots," in *International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 2022, pp. 709–714. DOI: 10.1109/ICARCV57592.2022.10004330.

[5] B. Sofman, E. Lin, J. A. Bagnell, J. Cole, N. Vandapel, and A. Stentz, "Improving robot navigation through self-supervised online learning," *Journal of Field Robotics*, vol. 23, no. 11-12, pp. 1059–1075, 2006. DOI: 10.1002/rob.20169.

[6] M. A. Bekhti, "Traversability cost prediction of outdoor terrains for mobile robot using image features," PhD thesis, Shizuoka University, Sep. 2020.

[7] V. Karolj, A. Viseras, L. Merino, and D. Shutin, "An integrated strategy for autonomous exploration of spatial processes in unknown environments," *Sensors*, vol. 20, no. 13, 2020. DOI: 10.3390/s20133663.

[8] J. Baleia, P. Santana, and J. Barata, "On exploiting haptic cues for self-supervised learning of depth-based robot navigation affordances," *Journal of Intelligent & Robotic Systems*, vol. 80, no. 3–4, pp. 455–474, 2015. DOI: 10.1007/s10846-015-0184-4.

[9] G. Kahn, P. Abbeel, and S. Levine, "BADGR: An autonomous self-supervised learning-based navigation system," *Robotics and Automation Letters*, vol. 6, no. 2, pp. 1312–1319, 2021. DOI: 10.1109/LRA.2021.3057023.

[10] P. Papadakis, "Terrain traversability analysis methods for unmanned ground vehicles: A survey," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 4, pp. 1373–1385, 2013. DOI: 10.1016/j.engappai.2013.01.006.

[11] M. Wermelinger, P. Fankhauser, R. Diethelm, P. Krüsi, R. Siegwart, and M. Hutter, "Navigation planning for legged robots in challenging terrain," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 1184–1189. DOI: 10.1109/IROS.2016.7759199.

[12] P. Krüsi, M. Bosse, and R. Siegwart, "Driving on point clouds: Motion planning, trajectory optimization, and terrain assessment in generic nonplanar environments," *Journal of Field Robotics*, vol. 34, no. 5, pp. 940–984, 2016. DOI: 10.1002/rob.21700.

[13] T. Homberger, M. Bjelonic, N. Kottege, and P. V. K. Borges, "Terrain-dependant Control of Hexapod Robots using Vision," in *International Symposium on Experimental Robotics (ISER)*, 2016, pp. 92–102. DOI: 10.1007/978-3-319-50115-4_9.

[14] J.-F. Lalonde, N. Vandapel, D. F. Huber, and M. Hebert, "Natural terrain classification using three-dimensional LADAR data for ground robot mobility," *Journal of Field Robotics*, vol. 23, no. 10, pp. 839–861, 2006. DOI: 10.1002/rob.20134.

[15] M. Kragh, R. N. Jørgensen, and H. Pedersen, "Object detection and terrain classification in agricultural fields using 3D lidar data," in *International Conference on Computer Vision Systems (ICVS)*, 2015, pp. 188–197. DOI: 10.1007/978-3-319-20904-3_18.

[16] D. Belter, J. Wietrzykowski, and P. Skrzypczyński, "Employing natural terrain semantics in motion planning for a multi-legged robot," *Journal of Intelligent & Robotic Systems*, vol. 93, no. 3, pp. 723–743, 2019. DOI: 10.1007/s10846-018-0865-x.

[17] M. Wigness, S. Eum, J. G. Rogers, D. Han, and H. Kwon, "A RUGD dataset for autonomous navigation and visual perception in unstructured outdoor environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 5000–5007. DOI: 10.1109/IROS40897.2019.8968283.

[18] P. Jiang, P. Osteen, M. Wigness, and S. Saripalli, "RELLIS-3D dataset: Data, benchmarks and analysis," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 1110–1116. DOI: 10.1109/ICRA48506.2021.9561251.

[19] A. Shaban, X. Meng, J. Lee, B. Boots, and D. Fox, "Semantic terrain classification for off-road autonomous driving," in *Conference on Robot Learning*, vol. 164, 2022, pp. 619–629.

[20] C. Cunningham, I. A. Nesnas, and W. L. Whittaker, "Improving slip prediction on Mars using thermal inertia measurements," *Autonomous Robots*, vol. 43, no. 2, pp. 503–521, 2019. DOI: 10.1007/s10514-018-9796-4.

[21] C. Ünsalan and K. L. Boyer, "Linearized vegetation indices based on a formal statistical framework," *Transactions on Geoscience and Remote Sensing*, vol. 42, no. 7, pp. 1575–1585, 2004. DOI: 10.1109/TGRS.2004.826787.

[22] J. Ahtiainen, T. Stoyanov, and J. Saarinen, "Normal distributions transform traversability maps: LIDAR-only approach for traversability mapping in outdoor environments," *Journal of Field Robotics*, vol. 34, no. 3, pp. 600–621, 2017. DOI: 10.1002/rob.21657.

[23] N. Kottege, C. Parkinson, P. Moghadam, A. Elfes, and S. P. N. Singh, "Energetics-informed hexapod gait transitions across terrains," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 5140–5147. DOI: 10.1109/ICRA.2015.7139915.

[24] V. A. Tucker, "The energetic cost of moving about: Walking and running are extremely inefficient forms of locomotion. Much greater efficiency is achieved by birds, fish and bicyclists," *American Scientist*, vol. 63, no. 4, pp. 413–419, 1975.

[25] W. Mou and A. Kleiner, "Online learning terrain classification for adaptive velocity control," in *IEEE Safety Security and Rescue Robotics (SSRR)*, 2010, pp. 1–7. DOI: 10.1109/SSRR.2010.5981563.

[26] K. Otsu, M. Ono, T. J. Fuchs, I. Baldwin, and T. Kubota, "Autonomous terrain classification with co- and self-training approach," *Robotics and Automation Letters*, vol. 1, no. 2, pp. 1–6, 2016. DOI: 10.1109/LRA.2016.2525040.

[27] R. B. McGhee and A. A. Frank, "On the stability properties of quadruped creeping gaits," *Mathematical Biosciences*, vol. 3, pp. 331–351, 1968. DOI: 10.1016/0025-5564(68)90090-4. (visited on 11/15/2018).

[28] B.-S. Lin and S.-M. Song, "Dynamic modeling, stability and energy efficiency of a quadrupedal walking machine," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 1993, pp. 657–670. DOI: 10.1109/ROBOT.1993.292201.

[29] S. Triest, M. G. Castro, P. Maheshwari, M. Sivaprakasam, W. Wang, and S. Scherer, "Learning risk-aware costmaps via inverse reinforcement learning for off-road navigation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 924–930. DOI: 10.1109/ICRA48891.2023.10161268.

[30] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, 27:1–27:27, 2011.

[31] V. da Silva and A. T. Winck, "Video popularity prediction in data streams based on context-independent features," in *Symposium on Applied Computing (SAC)*, 2017, pp. 95–100. DOI: 10.1145/3019612.3019638.

[32] R. C. Pinto and P. M. Engel, "A fast incremental Gaussian mixture model," *PLoS ONE*, vol. 10, no. 10, e0139931+, 2015. DOI: 10.1371/journal.pone.0139931.

[33] S. Vijayakumar and S. Schaal, "Locally weighted projection regression : An O(n) algorithm for incremental real time learning in high dimensional space," in *Intl. Conf. International Conference on Machine Learning (ICML)*, 2000, pp. 1079–1086.

[34] T. Kohonen, *Self-organizing Maps*, 3rd, ser. Springer series in information sciences, 30. Springer, 2001.

[35] B. Fritzke, "A growing neural gas network learns topologies," in *Conference on Neural Information Processing Systems (NIPS)*, 1994, pp. 625–632.

[36] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, ser. Adaptive Computation and Machine Learning. Cambridge, Mass: MIT Press, 2006.

[37] F. Shen, H. Yu, K. Sakurai, and O. Hasegawa, "An incremental online semi-supervised active learning algorithm based on self-organizing incremental neural network," *Neural Computing and Applications*, vol. 20, no. 7, pp. 1061–1074, 2011. DOI: 10.1007/s00521-010-0428-y.

[38] V. Tresp, "A Bayesian committee machine," *Neural Computation*, vol. 12, no. 11, pp. 2719–2741, 2000. DOI: 10.1162/089976600300014908.

[39] B. Yamauchi, "A frontier-based approach for autonomous exploration," in *IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA)*, 1997, pp. 146–151. DOI: 10.1109/CIRA.1997.613851.

[40] A. C. Schultz, W. Adams, and B. Yamauchi, "Integrating exploration, localization, navigation and planning with a common representation," *Autonomous Robots*, vol. 6, no. 3, pp. 293–308, 1999. DOI: 10.1023/A:1008936413435.

[41] F. Bourgault, A. A. Makarenko, S. B. Williams, B. Grocholsky, and H. F. Durrant-Whyte, "Information based adaptive robotic exploration," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 1, 2002, 540–545 vol.1. DOI: 10.1109/IRDS.2002.1041446.

[42] A. A. Makarenko, S. B. Williams, F. Bourgault, and H. F. Durrant-Whyte, "An experiment in integrated exploration," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 1, 2002, 534–539 vol.1. DOI: 10.1109/IRDS.2002.1041445.

[43] B. Charrow, S. Liu, V. Kumar, and N. Michael, "Information-theoretic mapping using Cauchy-Schwarz quadratic mutual information," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 4791–4798. DOI: 10.1109/ICRA.2015.7139865.

[44] C. Stachniss, G. Grisetti, and W. Burgard, "Information gain-based exploration using Rao-Blackwellized particle filters," in *Robotics: Science and Systems (RSS)*, 2005. DOI: 10.15607/RSS.2005.I.009.

[45] H. Carrillo, P. Dames, V. Kumar, and J. A. Castellanos, "Autonomous robotic exploration using a utility function based on Rényi's general theory of entropy," *Autonomous Robots*, vol. 42, no. 2, pp. 235–256, 2018. DOI: 10.1007/s10514-017-9662-9.

[46] W. Luo and K. Sycara, "Adaptive sampling and online learning in multi-robot sensor coverage with mixture of Gaussian processes," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 6359–6364. DOI: 10.1109/ICRA.2018.8460473.

[47] C. Rhodes, C. Liu, and W.-H. Chen, "Informative path planning for gas distribution mapping in cluttered environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 6726–6732. DOI: 10.1109/IROS45743.2020.9341781.

[48] K.-C. Ma, L. Liu, H. K. Heidarsson, and G. S. Sukhatme, "Data-driven learning and planning for environmental sampling," *Journal of Field Robotics*, vol. 35, no. 5, pp. 643–661, 2018. DOI: 10.1002/rob.21767.

[49] A. Singh, A. Krause, C. Guestrin, W. Kaiser, and M. Batalin, "Efficient planning of informative paths for multiple robots," in *International Joint Conference on Artifical Intelligence*, 2007, pp. 2204–2211.

[50] G. A. Hollinger and G. S. Sukhatme, "Sampling-based motion planning for robotic information gathering," in *Robotics: Science and Systems (RSS)*, 2013.

[51] J. Binney and G. S. Sukhatme, "Branch and bound for informative path planning," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 2147–2154. DOI: 10.1109/ICRA.2012.6224902.

[52] J. Faigl, M. Kulich, and L. Přeučil, "Goal assignment using distance cost in multi-robot exploration," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 3741–3746. DOI: 10.1109/IROS.2012.6385660.

[53] O. Mayuku, B. W. Surgenor, and J. A. Marshall, "A Self-supervised near-to-far approach for terrain-adaptive off-road autonomous driving," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 14 054–14 060. DOI: 10.1109/ICRA48506.2021.9562029.

[54] A. Rényi, "On measures of entropy and information," in *Berkeley Symposium on Mathematical Statistics and Probability*, 1961.

[55] C. E. Noon, "The generalized traveling salesman problem," PhD thesis, University of Michigan, 1988.

[56] Y. Prudent and A. Ennaji, "An incremental growing neural gas learns topologies," in *International Joint Conference on Neural Networks (IJCNN)*, vol. 2, 2005, pp. 1211–1216. DOI: 10.1109/IJCNN.2005.1556026.

Appendix A

Citations of Author's Publications

Listed publications of the Author, where equal contribution of all authors is assumed unless stated otherwise. For each publication, the number of citations based on Web of Science, Scopus, and Google Scholar is listed, extracted on November 28, 2023. For Web of Science and Scopus, self-citations are excluded. The citations extracted from the Web of Science are listed in detail.

A.1 Core Publications

A.1.1 Articles in Peer-reviewed Journals with Impact Factor

- M. Prágr *et al.*, "Autonomous exploration with online learning of traversable yet visually rigid obstacles," *Autonomous Robots*, vol. 47, pp. 161–180, 2023, Springer. DOI: 10.1007/s10514-022-10075-4. [C4].

  Listed as Q2 in Robotics with 3.5 Journal Impact Factor in 2022; citations: 0 in Web of Science, 1 in Scopus, 3 in Google Scholar.

A.1.2 Articles in Other Peer-reviewed Journals

- M. Prágr *et al.*, "Autonomous robotic exploration with simultaneous environment and traversability models learning," *Frontiers in Robotics and AI*, vol. 9, 2022. DOI: 10.3389/frobt.2022.910113. [C3].

  Citations: 1 in Web of Science, 1 in Scopus, 2 in Google Scholar; the citations listed in Web of Science follow.

  1. H. Wang *et al.*, "Design and locomotion analysis of an arm-wheel-track multimodal mobile robot," *Intelligent Service Robotics*, vol. 16, pp. 485–495, 2023. DOI: 10.1007/s11370-023-00472-8

A.1.3 Conference Proceedings Listed in Web of Science

- M. Prágr *et al.*, "Cost of transport estimation for legged robot based on terrain features inference from aerial scan," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 1745–1750. DOI: 10.1109/IROS.2018.8593374. [C1].

  Listed as A in CORE 2018; citations: 6 in Web of Science, 9 in Scopus, 22 in Google Scholar; the citations listed in Web of Science follow.

1. P. Ngamkajornwiwat *et al.*, "Bio-inspired adaptive locomotion control system for online adaptation of a walking robot on complex terrains," *IEEE Access*, vol. 8, pp. 91 587–91 602, 2020. DOI: `10.1109/ACCESS.2020.2992794`

2. J. Chen *et al.*, "Learning to identify footholds from geometric characteristics for a six-legged robot over rugged terrain," *Journal of Bionic Engineering*, vol. 17, no. 3, pp. 512–522, 2020. DOI: `10.1007/s42235-020-0041-4`

3. P. Arena *et al.*, "Energy efficiency of a quadruped robot with neuro-inspired control in complex environments," *Energies*, vol. 14, no. 2, 2021. DOI: `10.3390/en14020433`

4. Y. Chen *et al.*, "Sideways crab-walking is faster and more efficient than forward walking for a hexapod robot," *Bioinspiration & Biomimetics*, vol. 17, no. 4, 2022. DOI: `10.1088/1748-3190/ac6847`

5. S. Sun *et al.*, "A novel adaptive methodology for removing spurious components in a modified incremental gaussian mixture model," *International Journal of Machine Learning and Cybernetics*, vol. 14, no. 2, pp. 551–566, 2023. DOI: `10.1007/s13042-022-01649-w`

6. H. Azpurua *et al.*, "A survey on the autonomous exploration of confined subterranean spaces: Perspectives from real-word and industrial robotic deployments," *Robotics and Autonomous Systems*, vol. 160, 2023. DOI: `10.1016/j.robot.2022.104304`

- M. Prágr *et al.*, "Online incremental learning of the terrain traversal cost in autonomous exploration," in *Robotics: Science and Systems (RSS)*, 2019. DOI: `10.15607/RSS.2019.XV.040`. [C2].

Listed as A* in CORE 2018; citations: 5 in Web of Science, 5 in Scopus, 22 in Google Scholar; the citations listed in Web of Science follow.

1. V. Karolj *et al.*, "An integrated strategy for autonomous exploration of spatial processes in unknown environments," *Sensors*, vol. 20, no. 13, 2020. DOI: `10.3390/s20133663`

2. H. Azpurua *et al.*, "Three-dimensional terrain aware autonomous exploration for subterranean and confined spaces," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 2443–2449. DOI: `10.1109/ICRA48506.2021.9561099`

3. M. Sivaprakasam *et al.*, "Improving off-road planning techniques with learned costs from physical interactions," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 4844–4850. DOI: `10.1109/ICRA48506.2021.9561881`

4. A. Kurobe *et al.*, "Audio-visual self-supervised terrain type recognition for ground mobile platforms," *IEEE Access*, vol. 9, pp. 29 970–29 979, 2021. DOI: `10.1109/ACCESS.2021.3059620`

5. H. Azpurua *et al.*, "A survey on the autonomous exploration of confined subterranean spaces: Perspectives from real-word and industrial robotic deployments," *Robotics and Autonomous Systems*, vol. 160, 2023. DOI: `10.1016/j.robot.2022.104304`

A.2 Related Non-core Publications

A.2.1 Conference Proceedings Listed in Web of Science

- J. Faigl and M. Prágr, "On unsupervised learning of traversal cost and terrain types identification using self-organizing maps," in *International Conference on Artificial Neural Networks (ICANN)*, Springer, 2019, pp. 654–668. DOI: `10.1007/978-3-030-30487-4_50`. [R4].

  Listed as B in CORE 2018; citations: 3 in Web of Science, 5 in Scopus, 9 in Google Scholar; the citations listed in Web of Science follow.

  1. D. C. Guastella and G. Muscato, "Learning-based methods of perception and navigation for ground vehicles in unstructured environments: A review," *Sensors*, vol. 21, no. 1, 2021. DOI: `10.3390/s21010073`

  2. L. Gan *et al.*, "Energy-based legged robots terrain traversability modeling via deep inverse reinforcement learning," *Robotics and Automation Letters*, vol. 7, no. 4, pp. 8807–8814, 2022. DOI: `10.1109/LRA.2022.3188100`

  3. P. Xu *et al.*, "Learning physical characteristics like animals for legged robots," *National Science Review*, vol. 10, no. 5, 2023. DOI: `10.1093/nsr/nwad045`

- M. Prágr *et al.*, "Incremental learning of traversability cost for aerial reconnaissance support to ground units," in *2018 Modeling & Simulation for Autonomous Systems (MESAS)*, Springer, 2019, pp. 412–421. DOI: `10.1007/978-3-030-14984-0_30`. [R1].

  Citations: 1 in Web of Science, 1 in Scopus, 9 in Google Scholar; the citations listed in Web of Science follow.

  1. S. Sun *et al.*, "A novel adaptive methodology for removing spurious components in a modified incremental gaussian mixture model," *International Journal of Machine Learning and Cybernetics*, vol. 14, no. 2, pp. 551–566, 2023. DOI: `10.1007/s13042-022-01649-w`

- M. Prágr and J. Faigl, "Benchmarking incremental regressors in traversal cost assessment," in *International Conference on Artificial Neural Networks (ICANN)*, Springer, 2019, pp. 685–697. DOI: `10.1007/978-3-030-30487-4_52`. [R5].

  Listed as B in CORE 2018; citations: 0 in Web of Science, 0 in Scopus, 3 in Google Scholar.

- M. Prágr *et al.*, "Traversal cost modeling based on motion characterization for multi-legged walking robots," in *European Conference on Mobile Robots (ECMR)*, IEEE, 2019, pp. 1–6. DOI: `10.1109/ECMR.2019.8870912`. [R10].

  Citations: 0 in Web of Science, 1 in Scopus, 2 in Google Scholar.

- M. Prágr and J. Faigl, "Terrain learning using time series of ground unit traversal cost," in *2019 Modeling & Simulation for Autonomous Systems (MESAS)*, Springer, 2020, pp. 97–107. DOI: `10.1007/978-3-030-43890-6_8`. [R11].

  Citations: 0 in Web of Science, 0 in Scopus, 1 in Google Scholar.

- M. Prágr *et al.*, "Aerial reconnaissance and ground robot terrain learning in traversal cost assessment," in *2019 Modeling & Simulation for Autonomous Systems (MESAS)*, Springer, 2020, pp. 3–10. DOI: `10.1007/978-3-030-43890-6_1`. [R2].

Citations: 1 in Web of Science, 2 in Scopus, 3 in Google Scholar; the citations listed in Web of Science follow.

1. C. D. Braga Borges and J. J. S. J. de Mesquita, "Traversability learning from aerial images with fully convolutional neural networks," *Neural Processing Letters*, 2023. DOI: `10.1007/s11063-023-11406-x`

- R. Szadkowski *et al.*, "Transfer of inter-robotic inductive classifier," in *International Conference on Automation, Control and Robots (ICACR)*, IEEE, 2020, pp. 32–36. DOI: `10.1109/ICACR51161.2020.9265509`. [R6].

Citations: 0 in Web of Science, 0 in Scopus, 1 in Google Scholar.

- M. Zoula *et al.*, "On building communication maps in subterranean environments," in *2020 Modeling & Simulation for Autonomous Systems (MESAS)*, Springer, 2021, pp. 15–28. DOI: `10.1007/978-3-030-70740-8_2`. [R12].

Citations: 0 in Web of Science, 0 in Scopus, 2 in Google Scholar.

- J. Zelinka *et al.*, "Traversability transfer learning between robots with different cost assessment policies," in *2021 Modeling & Simulation for Autonomous Systems (MESAS)*, Springer, vol. 13207, 2022, pp. 333–344. DOI: `10.1007/978-3-030-98260-7_21`. [R7].

Citations: 0 in Web of Science, 0 in Scopus, 2 in Google Scholar.

- M. Prágr *et al.*, "Terrain traversal cost learning with knowledge transfer between multi-legged walking robot gaits," in *IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2022, pp. 148–153. DOI: `10.1109/ICARSC55462.2022.9784790`. [R8].

Citations: 1 in Web of Science, 1 in Scopus, 1 in Google Scholar; the citations listed in Web of Science follow.

1. M. Eder *et al.*, "Traversability analysis for off-road environments using locomotion experiments and Earth observation data," *Robotics and Autonomous Systems*, vol. 168, p. 104 494, 2023. DOI: `10.1016/j.robot.2023.104494`

A.2.2   Other Conference Proceedings

- J. Faigl and M. Prágr, "Incremental traversability assessment learning using growing neural gas algorithm," in *Advances in Self-Organizing Maps, Learning Vector Quantization, Clustering and Data Visualization (WSOM 2019)*, Springer, 2020, pp. 166–176. DOI: `10.1007/978-3-030-19642-4_17`. [R3].

Citations: 0 in Web of Science, 5 in Scopus, 9 in Google Scholar.

- J. Zelinka *et al.*, "Deep transfer learning of traversability assessment for heterogeneous robots," in *Conference Information Technologies - Applications and Theory (ITAT)*, vol. 3226, 2022, pp. 12–20. [R9].

Citations: 0 in Web of Science, 0 in Scopus, 0 in Google Scholar.

A.3 Other Publications

A.3.1 Articles in Peer-reviewed Journals with Impact Factor

- R. Szadkowski *et al.*, "Self-learning event mistiming detector based on central pattern generator," *Frontiers in Neurorobotics*, vol. 15, p. 5, 2021. DOI: 10.3389/fnbot.2021. 629652.

Listed as Q2 in Computer Science, Artificial Intelligence with 2.6 Journal Impact Factor in 2020; citations: 5 in Web of Science, 5 in Scopus, 6 in Google Scholar; the citations listed in Web of Science follow.

1. P. Manoonpong *et al.*, "Insect-inspired robots: Bridging biological and artificial systems," *Sensors*, vol. 21, no. 22, 2021. DOI: 10.3390/s21227609

2. A. Srisuchinnawong *et al.*, "Neurovis: Real-time neural information measurement and visualization of embodied neural systems," *Frontiers in Neural Circuits*, vol. 15, 2021. DOI: 10.3389/fncir.2021.743101

3. P. Arena *et al.*, "Ground reaction force estimation in a quadruped robot via liquid state networks," in *International Joint Conference on Neural Networks (IJCNN)*, 2022. DOI: 10.1109/IJCNN55064.2022.9892423

4. D. Owaki *et al.*, "Editorial: Biological and robotic inter-limb coordination," *Frontiers in Robotics and AI*, vol. 9, 2022. DOI: 10.3389/frobt.2022.875493

5. S. Li *et al.*, "A protocol to analyze the global literature on the clinical benefit of interlimb-coordinated intervention in gait recovery and the associated neurophysiological changes in patients with stroke," *Frontiers in Neurology*, vol. 13, 2022. DOI: 10.3389/fneur.2022.959917

A.3.2 Other Conference Proceedings

- J. Faigl *et al.*, "On Autonomous mobile robot exploration projects in robotics course," in *International Conference on Robotics in Education (RiE)*, 2023, pp. 3–15. DOI: 10. 1007/978-3-031-38454-7_1.

Citations: 0 in Web of Science, 0 in Scopus, 0 in Google Scholar.

Appendix B

Full Texts of the Core Publications

Cost of Transport Estimation for Legged Robot Based on Terrain Features Inference from Aerial Scan

Miloš Prágr   Petr Čížek   Jan Faigl

Abstract— The effectiveness of the robot locomotion can be measured using the cost of transport (CoT) which represents the amount of energy that is needed for traversing from one place to another. Terrains excerpt different mechanical properties when crawled by a multi-legged robot, and thus different values of the CoT. It is therefore desirable to estimate the CoT in advance and plan the robot motion accordingly. However, the CoT might not be known prior the robot deployment, e.g., in extraterrestrial missions; hence, a robot has to learn different terrains as it crawls through the environment incrementally. In this work, we focus on estimating the CoT from visual and geometrical data of the crawled terrain. A thorough analysis of different terrain descriptors within the context of incremental learning is presented to select the best performing approach. We report on the achieved results and experimental verification of the selected approaches with a real hexapod robot crawling over six different terrains.

I. INTRODUCTION

Autonomous robots are being deployed in long-term data collection missions in environments with limited or no prior information about the particular terrain the robots are facing to, e.g., in extraterrestrial missions [1]. However, efficient locomotion over a particular terrain greatly influences the mission effectiveness. It is even more prominent with multi-legged robots due to their enhanced traversability capabilities to reside over terrains of different types.

Regarding locomotion of the particular robot, terrains can be distinguished by the traversability cost metric [2] that can be a simple binary division between passable and impassable terrains [3]. Alternatively, there are more elaborating scores such as the Cost of Transport (CoT) [4], [5], which represents a measure of the effectiveness of the robot locomotion. Therefore it is desirable to study the terrain traversability estimation to support mission planning and improve the real-time robot performance in accomplishing its mission goals.

Note, the CoT is inherently a continuous measure influenced by many factors, e.g., terramechanical properties of the terrain, robot morphology, and even seasonal and weather condition changes in long-term missions. Besides, terrains that the robot encounters might not be known in advance, and therefore, a simple classification using a set of pre-learned classes to estimate the CoT is not sufficient for a real-world deployment. Hence, self-learning mechanisms are necessary to estimate the CoT in yet untraversed areas correctly.

Authors are with the Department of Computer Science, Faculty of Electrical Engineering, Czech Technical University, Prague, Czech Republic. {pragrmi1, petr.cizek, faiglj}@fel.cvut.cz

(a) CoT estimates

(b) Colored height map   (c) Multiple artificial terrains
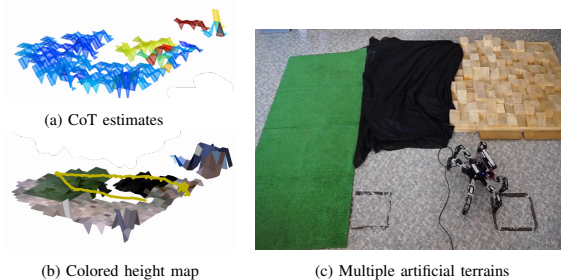
Fig. 1: Multi-Terrain setup and its perceived representation.

In this work, we are concerning the traversability cost estimation for a small hexapod crawling robot using extero-ceptive data. A relatively slow speed of crawling robots limits their capability to map a given area fully. Moreover, an easily obstructive close-to-ground viewpoint makes the mapping even more challenging. Nevertheless, a robot capable of traversing an unknown terrain may observe both the terrain appearance and terramechanical properties, and thus it can incrementally build the terrain model describing its CoT.

On the other hand, unmanned aerial vehicles (UAVs) do not suffer from these problems. From a relatively high altitude, a UAV can observe a larger portion of the terrain. Also, a typical UAV often moves faster than most of the ground vehicles of similar size. For these reasons, UAVs can be utilized for mapping unknown terrains, which in turn, may help the robot to improve the mission efficiency.

However, the ground crawling robot and the UAV observe the terrain with a different perspective, and therefore, it is not possible to simply infer the CoT over the terrain observed by the UAV based on the model learned by the ground robots if arbitrary descriptors are used. Hence, construction of such a model together with a selection of suitable feature set for the desired inference is the main goal of this work. In particular, we aim to develop incremental learning of the model to estimate the CoT based on the experience of the hexapod crawling robot with the terrain that can be further utilized for inference of the CoT using aerial terrain view. Such an inference may lead to the annotated aerial terrain cost map as it is visualized in Fig. 1. In this paper, we report on the achieved results towards this challenging goal which is addressed as a thorough experimental analysis of a set of terrain characterization features and learning methods used in the literature for terrain classification from visual and scene geometric data and application of these methods in the problem of the CoT learning and estimation.

The paper is structured as follows. An overview of the

related approaches on the terrain classification and terrain features is in Section II. A description of the proposed inference learning framework, terrain model, and features utilized in the herein reported evaluation is presented in Section III. Section IV details the experimental set-up and reports on the achieved results. Finally, conclusions are drawn in Section V.

II. RELATED WORK

Numerous methods on terrain traversability analysis [2], [6] have been presented in recent years that aim to evaluate the terrain properties mostly from geometric data and assign each area a number that characterizes local properties of the terrain geometry. On the other hand, traversability cost metrics [4], [5], like the Cost of Transport (CoT), are more general metrics that incorporate the own robot experience with traversing the terrain. The value of the CoT is inherently continuous. Even though it can be estimated from the results of the terrain classification, the classification relies on a discrete set of pre-learned classes which might not be available or might get irrelevant during the mission, e.g., in extraterrestrial environments or long-term missions. Therefore we are interested in the CoT regression.

The terrain description methods rely on extraction of terrain characterization features that can be roughly categorized into appearance-based visual features, geometric-based features, and methods combining both approaches. Moreover, the approaches can be further categorized based on whether the feature is dependent on the current robot viewpoint, and whether the feature makes use of color information. For example, approaches extracting features from images are inherently viewpoint dependent. In contrast, approaches using the extraction of features from point clouds or aerial scans are position independent. Further, the color information can be valuable for discriminating terrain types; however, it is strongly influenced by the illumination and seasonal changes.

The appearance-based features include approaches to classify terrains based on different colors or textures. The authors of [7] introduce classification based on a simple two-dimensional feature which uses mean color components of superpixels in the Lab color space combined with the SVM classifier. Bayes decision rules with Gaussian mixture models on the RGB color space is used in [8]. Regarding the texture recognition, methods using frequency-based approaches relying on wavelet filters [9], [10], Gabor filters [11], or more recent approach on Steerable Pyramid Masks [12] can be used. A survey on visual terrain classification from a monocular camera is presented in [13]. Further, hybrid approaches utilizing RGB-D camera [14] and stereo camera [15] use a combination of the appearance and geometrical features.

The main advantage of the geometrical terrain features is that they are not affected by illumination changes. However, they can suffer from a low density of the point cloud in a far distance to the robot which is limiting especially for a vast number of approaches that compute statistics based on normals [16]–[19] where point normals are mostly computed by fitting a plane to a local neighborhood of the investigated

points. Point-cloud density [20], view-point, and point cloud centroid relations [18], [19] or minimal and maximal curvature of estimates for local neighborhood [21], or histogram based features [22] are being used. Features extracted from LIDAR data are used in [23] with a random forest classifier to differentiate vegetation and estimate the soil plane using the geometrical, reflectance, and color information.

In [24], a set of 13 features is proposed to describe terrain, vegetation, and other objects in an agricultural environment. The set is computed from the local neighborhood of the interesting point and divided into four height features based on the z coordinate, four shape features based on principal component analysis, three orientation features based on normal vectors of the local plane, a distance feature, and a reflectance feature. However, the classifier is trained by SVM from labeled data [24]. Moreover, the approach requires the z-coordinate of the point cloud to be orthogonal to the surface which is made by fitting a global ground plane to the dataset, which may bias the results in a more structured environment.

A self-supervised approach is presented in [25] to teach a terrain classifier from geometric data using the proprioceptive data. However, the approach uses a pre-learned proprioceptive classifier which differentiates only between several terrain classes.

The most similar approach to the herein addressed problem of the assignment of the traversability data to the aerial scan has been presented in [20]. The authors proposed a self-supervised learning approach with a Gaussian mixture model. The traversability cost is estimated from the geometrical clues in the LIDAR data to infer the traversability cost in a map obtained by an aerial recon.

Based on the presented literature survey, we have identified a set of appearance-based and geometric-based features with different properties that are commonly used in the terrain classification, and we adapt these for the CoT regression presented in this paper. Description of the used features is presented in the following section. Besides, we can conclude that the incremental regression and estimation of the traversability value for the observed but yet untraversed areas is still a largely unexplored topic.

III. INFERENCE LEARNING FRAMEWORK

The main goal of this paper is to report on a thorough analysis of terrain characterization features used in the inference of the CoT perceived by the hexapod crawling robot to the aerial scan. This section describes the used framework for the extraction of the terrain features and CoT learning, i.e., the utilized terrain features, the learning procedure, and the sampling strategy for the inference learning. The individual building blocks are described in the following sections.

A. Terrain Characterization Features

Based on the literature survey and preliminary results, we consider the following features for benchmarking. We select the features that are computationally inexpensive, so that can be utilized on various mobile robotic platforms. Moreover,

we only use descriptors that are viewpoint robust under the herein described conditions.

1) Appearance-based features: Two point cloud based color features and a texture recognition using wavelets [9] have been selected. For the color features, both the RGB and Lab color spaces have been considered with either channel values of the sampled point (denoted as Point in the reported results) or a channel mean of the points in a $r = 0.2$ m spherical neighborhood (denoted as Mean in the reported results). As the feature is purely based on color, it is robust to viewpoint changes; however, less to illumination changes. During the preliminary evaluation, the wavelet features exhibit low performance presumably due to a large viewpoint change between the robot and the aerial scan, and they have been left out of the comparison.

2) Geometric-based features: We use a modified version of the terrain feature sets presented in [24]. In particular, we have used 11 out of 13 features, namely the shape feature, height feature, orientation feature and all of them combined in a full feature, leaving out the reflectance and distance features as those do not suit our experimental setup. The ground plane and normal are estimated by fitting a plane to the $k = 5$ nearest neighbors of the sampled point. We consider the utilized geometric features to be viewpoint robust. As the coordinate frame is based on the global ground plane estimate, the height feature and shape features are robust to viewpoint changes under the assumption that the aerial scan captures the area with sufficient precision. The orientation feature robustness depends on the quality of the aerial scan, although different descriptor values are assigned to the terrains sloped in different directions. A spherical region with $r = 0.3$ m radius is considered when querying the neighborhood of the sampled point.

B. Learning algorithms

We have considered four approaches on top of the utilized terrain features that are capable of regression from which two of them support incremental online learning.

1) Support Vector Regression: (SVR) [26] is a maximum-margin regression algorithm, here utilized with the radial basis function kernel.

2) Regression Tree: which uses recursive partitioning with the depth $d = 5$.

3) Incremental Gaussian Mixture Network Model: (IGMN) [27], [28] is an online incremental learning approach, which creates and updates the Gaussian mixture model based on streamed data points. The IGMN allows a full prediction of the data point based on an incomplete input of any kind. We used our implementation of the Fast-IGMN that is an improvement of the IGMN presented in [28]. The Fast-IGMN improves the IGMN time complexity to $\mathcal{O}(NKD^2)$, where N is the number of data points, K is the number of components, and D is the data point dimensionality. Experimentally, we parametrized the IGMN with the $k = 10$ components, grace period $v_{\min} = 100$, minimal accumulated posterior $sp_{\min} = 3$, and scaling factor $\delta = 1$.

4) Hoeffding Tree: or Very Fast Decision Tree Learner (VFDT) [29] is an online incremental decision tree learning algorithm that utilizes Hoeffding bound to create the output asymptotically identical to that of the conventional learner. We used a slightly modified VFDT implementation of [30]. However, unlike the other utilized approaches, Hoeffding tree is used with a discrete number of classes, i.e., $k = 10$.

C. Sampling and Learning

This section explains how the above-selected terrain features and learning algorithms (forming building blocks) are combined in an inference learning framework that estimates the CoT in the aerial scan of the terrain. Our model has two major life stages: (i) the learning phase, when the robot learns the model based on the RGB-D input and pairs it with the CoT; and (ii) the inference phase, where the learned model is used to evaluate the terrain observed from the aerial scan.

The framework operates on individual datasets consisting of georeferenced RGB-D, i.e., the color RGB and depth images from the robot and georeferenced RGB-D aerial scan of the whole environment. Besides, the ground robot collects the power readings used for estimation of the CoT [5] as

$$\text{CoT} = \frac{P}{m\,g\,v}, \tag{1}$$

where P is the instantaneous power consumption, m is the weight of the robot, $g = 9.81$ ms^{-2} is the gravitational acceleration, and v is the robot speed. In the regression task, we understand the CoT to be a function of the robot type, the robot gait, and the local terrain property. However, the experimental platform and locomotion gait are fixed; hence, the CoT is estimated only from the local terrain property. Note that the evaluated aerial scan is independent of the robot trajectory, i.e., the robot learned model can be applied to a different location; however, for the herein presented benchmarking, it is necessary that the trajectory of the robot is contained within the aerial scan.

It is essential to address the fact that the robot knows the CoT only after it successfully traverses the terrain and estimates its velocity from the georeferenced data. Therefore, we introduce feature storage that maintains a dictionary of the georeferenced features extracted from the robot field of view to deal with this delay in the acquisition of the RGB-D–CoT pairs that are necessary for the learning phase. Thus, whenever an arbitrary location is reached by the robot, the feature storage is queried, and all features located sufficiently close, i.e., in a spherical region with $r = 0.2$ m, are passed to the learning framework together with the measured CoT of the current location. The temporary feature storage is of limited size, and it is randomly pruned when its capacity overflows. Hence, the robot is not creating a persistent feature map of the environment that would grow over time, and it incrementally learns the forthcoming terrain.

During the learning phase, points in front of the robot are sampled according to the scheme described above. The obtained features are then used for learning of the CoT model. In the case of the incremental learning, the features are first used to query the model for the value of the CoT.

(a) The rough terrains.

(b) CoT estimation for different variants of appearence-based features using the IGMN.

(c) CoT estimation for different variants of geometric-based features using the IGMN.

(d) Comparison of CoT estimation using the RGB-M-Shape feature and different learning algorithms.

(e) Comparison of CoT estimation for different variants of combined appearance and geometric features using the IGMN.

(f) Comparison of CoT estimation for different variants of combined appearance and geometric features using Regression Tree.

(b-f) Learned and predicted values of the CoT in time for different combination of features and learning algorithms. The first six terrains (parts) represent the learning phase. The following six parts represent the inference phase.

(g) The used hexapod robot.

(h) Mean absolute error and its variance of selected features. Variance bars not to scale.

Fig. 2: Experimental Hardware, Setup, and Results.

For the inference phase, the robot is virtually walked along the same known trajectory while sampling points from the aerial scan, which is necessary for the evaluation purposes. The robot stores the geolocated features in the same manner as in the learning phase. Similarly, when an arbitrary location is reached, the feature storage is queried, and all features located sufficiently close are considered. These features are then used to query the CoT model for the particular value of the CoT. Additionally, for the incremental learning enabled approaches, the model is further taught by these features combined with the measured CoT.

IV. EXPERIMENTAL EVALUATION

In this section, we report on the experimental results and verification of the proposed inference learning framework with terrain features benchmarking using real hexapod crawling robot. The reported results are organized as follows. First, the robotic platform and the experimental set-up are introduced. After that, the results themselves are presented and discussed.

A. Hexapod Crawling Robot

The used robot is an electrically actuated low-cost hexapod crawling robot depicted in Fig. 2g. It has six legs, each with three joints attached to the trunk which hosts the electronics and sensory equipment. The RGB-D ASUS Xtion Pro Live camera has been utilized for the terrain perception and the Hall-effect-based current sensor for estimation of the robot instantaneous power consumption. The camera provides the data with 30 Hz frequency, and the power consumption data are provided with 62 Hz frequency. The locomotion over the rough terrain is performed by the adaptive motion gait [31], which uses the estimation of the ground-reaction forces based on the position data provided by the joint actuators.

B. Experimental Setup and Terrains

The experimental data have been obtained on the laboratory test-track consisting of three-meter length path over different surfaces. Six experiments, each with different terrain and three trials, have been performed. The terrains with the increasing difficulty of traversing are: PVC flooring (flat),

TABLE I: Error Rates for Individual Terrain Characterization Features

Learning	Feature Color	Geom.	Error Mean	Var	Learning	Feature Color	Geom.	Error Mean	Var	Learning	Feature Color	Geom.	Error Mean	Var
Hoeffding	RGB-M	Shape	7.4	69.8	IGMN	None	Ori	8.7	57.4	Reg Tree	Lab-P	None	6.4	37.9
Hoeffding	None	Height	7.4	56.7	IGMN	RGB-M	None	5.7	44.5	Reg Tree	Lab-P	Shape	7.1	70.9
Hoeffding	Lab-M	None	8.2	86.1	IGMN	RGB-M	Height	6.6	39.0	Reg Tree	None	Ori	8.1	60.3
Hoeffding	Lab-M	Shape	8.6	77.0	IGMN	RGB-M	Ori	5.8	43.6	Reg Tree	None	Set	7.9	59.0
Hoeffding	Lab-P	None	8.9	102.2	IGMN	RGB-M	Set	7.8	38.9	Reg Tree	None	Shape	6.9	77.2
Hoeffding	Lab-P	Shape	7.7	57.3	IGMN	RGB-M	Shape	**5.7**	**38.4**	SVR	RGB-M	Shape	6.8	76.3
Hoeffding	None	Ori	9.0	112.2	IGMN	RGB-P	None	7.3	59.4	SVR	None	Height	7.8	73.9
Hoeffding	RGB-M	None	8.1	70.1	IGMN	RGB-P	Shape	6.7	40.4	SVR	Lab-M	None	7.1	81.5
Hoeffding	RGB-M	Height	7.4	59.8	IGMN	None	Set	9.0	58.5	SVR	Lab-M	Shape	7.9	64.9
Hoeffding	RGB-M	Ori	7.9	66.0	IGMN	None	Shape	8.4	44.9	SVR	Lab-P	None	6.7	63.8
Hoeffding	RGB-M	Set	9.3	87.9	Reg Tree	RGB-M	None	6.3	51.5	SVR	Lab-P	Shape	7.8	68.7
Hoeffding	RGB-P	None	7.7	63.5	Reg Tree	RGB-M	Height	6.4	48.4	SVR	None	Ori	7.6	79.9
Hoeffding	RGB-P	Shape	8.3	72.5	Reg Tree	RGB-M	Ori	5.8	47.5	SVR	RGB-M	None	7.2	74.8
Hoeffding	None	Set	8.6	76.2	Reg Tree	RGB-M	Set	6.8	63.3	SVR	RGB-M	Height	7.7	74.0
Hoeffding	None	Shape	9.1	82.5	Reg Tree	RGB-M	Shape	5.9	49.6	SVR	RGB-M	Ori	7.4	77.2
IGMN	None	Height	9.0	37.9	Reg Tree	None	Height	7.3	58.6	SVR	RGB-M	Set	7.7	78.1
IGMN	Lab-M	None	**5.7**	**38.3**	Reg Tree	RGB-P	None	6.2	41.5	SVR	RGB-P	None	7.0	75.5
IGMN	Lab-M	Shape	9.4	147.2	Reg Tree	Lab-M	None	6.9	66.6	SVR	RGB-P	Shape	6.9	70.7
IGMN	Lab-P	None	7.5	39.6	Reg Tree	RGB-P	Shape	6.6	58.7	SVR	None	Set	7.7	78.3
IGMN	Lab-P	Shape	7.2	67.6	Reg Tree	Lab-M	Shape	6.1	46.0	SVR	None	Shape	7.3	73.7

turf-like carpet (grass), and semi-transparent soft black fabric (black) represent different flat terrains. Then, wooden blocks covered with the turf-like carpet (grass rough), wooden blocks covered with the black fabric (black rough), and bare wooden blocks (blocks) are considered as the rough terrain scenarios. The wooden blocks are 10×10 cm large with variable height and slope. The three rough terrain setups are shown in Fig. 2a. The same turf-like carpet and black-fabric have been used for the flat and the rough terrain setups.

The robot has been remotely guided over the course of the test-track while collecting visual and power consumption data. The visual data have been then processed using the incremental localization technique [32] to extract localization information that has been further used to estimate the robot velocity and for calculation of the CoT according to (1).

The collected data represent an unbiased belief of the robot about the traversability of the selected terrains. Besides, an aerial scan has been captured for each terrain type from the elevated camera to allow dense reconstruction of the whole track course simulating an aerial scan.

C. Results and Discussion

First, the individual trials over different terrains have been merged into a single pass dataset with six terrains each repeated two times as it is visualized in Figs. 2b-2f. Then, the proposed framework has been used according to the description presented in Section III-C. The algorithm has learned on the first six terrains, whereas the CoT value for the following terrains has been inferred from the aerial scan. The incremental learning approaches learn from, but also return the evaluation, for all the terrains, i.e., the terrains used in both the learning and inference phases.

The quantitative measures of the mean error between the predicted and ground-truth CoT, and its variance are reported for all tested combinations in Table I. Additionally, some of the results are visualized in Fig. 2h. Note that the measure is computed from all the returned values and for the incremental learning approaches, the metric includes results

TABLE II: Walk- and Environment-scan Correlation Rates for the RGB Mean Shape feature. For each of the examined terrain type pairs the metric is computed from 1000 randomly selected points. Each point is represented by the RGB-Means Shape descriptor computed from the environemnt scan, and the same descriptor computed from the walk scan. Then each dimension of the descriptor is reported separately, i.e., the correlation for one dimension in one terrain type is computed from 2 1000-length vectors. A median of the individual dimension correlations is also reported.

Terrain Type	Individual Feature Building Blocks Shape				RGB Means			Full Feature Median
Grass Flat	0.62	0.40	0.34	0.66	0.66	0.47	0.64	0.62
Grass Rough	0.31	0.33	0.29	0.19	0.78	0.75	0.58	0.33
Black Flat	0.56	0.46	0.42	0.53	0.86	0.85	0.84	0.56
Black Rough	0.36	0.48	0.38	0.36	0.74	0.75	0.75	0.48
Flat	0.63	0.51	0.43	0.78	0.09	0.03	0.08	0.43
Cubes	0.55	0.34	0.31	0.47	0.79	0.80	0.78	0.55

returned on the first six learning terrains.

The preliminary analysis has shown a low quality of the results provided by the Hoeffding trees and SVR learning algorithms. Therefore, the qualitative evaluation is focused on the best performing terrain features using the IGMN and Regression Trees. From the quantitative comparison, the best performing features are the sole appearance-based the LAB Mean feature and RGB Mean feature together with the geometric-based Shape feature.

A good performance of the sole Lab Mean feature (see Fig. 2b) is not surprising partially because of the experimental setup where the four well distinguishable colors appear on the terrains. Besides, only a little difference between the CoT values for three out of six terrains, namely flat, grass and grass rough, has been observed, which is correct behavior as we are not interested in the terrain classification, but rather in the CoT estimation. The CoT over the wooden blocks is less uniform, with low-cost areas being similar to the flat or grass datasets and high-cost peaks. Finally, the black and rough black datasets are the most costly with the high-cost peaks. Presumably, it is caused by the inability of the robot to find a proper grip on the fabric covered terrain. The made observations comply with the CoT map presented in Fig. 1, where the blocks have assigned a range of different costs, whereas the black fabric is assigned the high costs only.

From the further qualitative analysis, we can see that

the standalone geometric features do not perform well (see Fig. 2c). However, the combination of the color and shape feature provides, in our opinion, the best results as the combination is able to better cope with the high peaks and low values of the CoT. In Table II, we present the aerial-ground scan correlation of the RGB Mean Shape feature. Although the individual feature dimensions do not exhibit a high correlation for all the terrains, for each terrain, there is at least one dimension with considerable correlation. The comparison of different models favors the IGMN setup in both quantitative and qualitative measures (see Fig. 2d, Fig. 2e, and Fig. 2f). It is most likely due to its incremental learning property that allows the model to adapt quickly to CoT changes.

A rather interesting property of our datasets is that at the far end of each examined terrain, there is usually a section of a flat ground that has been traversed by the robot. Such a border represents a change of the terrain type. When investigating the recovered data, it is possible to observe that the most of the good performing setups are capable of reacting on such a terrain change and presume a lower value of the CoT in that region.

V. Conclusion

In this paper, we present a framework for model learning the CoT in a two-viewpoint setups, where the model is firstly learned by a small ground hexapod crawling robot, which can observe not only the exteroceptive terrain properties but also its associated CoT, and then the model is used for CoT inference from an aerial scan to yet untraversed areas. From a set of several feature setups, we chose the best performing combination of the RGB Mean and Shape feature which forms the descriptor of only seven dimensions. Several learning setups have been evaluated and based on the achieved results, the incremental Gaussian mixture and post-estimation regression trees suit best the selected features. In future, we aim to utilize this system in planning tasks and explore the transferability of the terrain traversability evaluation between different robotic platforms.

References

[1] D. Brown and G. Webster, "Now a stationary research platform, nasa's mars rover spirit starts a new chapter in red planet scientific studies," *NASA, press release*, 2010.

[2] P. Papadakis, "Terrain traversability analysis methods for unmanned ground vehicles: A survey," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 4, pp. 1373–1385, 2013.

[3] A. Stelzer, H. Hirschmüller, and M. Görner, "Stereo-vision-based navigation of a six-legged walking robot in unknown rough terrain," *The International Journal of Robotics Research*, vol. 31, no. 4, pp. 381–402, 2012.

[4] J. Nishii, "An analytical estimation of the energy cost for legged locomotion," *Journal of theoretical biology*, vol. 238, no. 3, pp. 636–645, 2006.

[5] N. Kottege, C. Parkinson, P. Moghadam, A. Elfes, and S. P. Singh, "Energetics-informed hexapod gait transitions across terrains," in *ICRA*, 2015, pp. 5140–5147.

[6] G. Reina, M. Bellone, L. Spedicato, and N. I. Giannoccaro, "3D traversability awareness for rough terrain mobile robots," *Sensor Review*, vol. 34, no. 2, pp. 220–232, 2014.

[7] K. Otsu, M. Ono, T. J. Fuchs, I. Baldwin, and T. Kubota, "Autonomous Terrain Classification with Co- and Self-Training Approach," in *IEEE Robotics and Automation Letters*, 2016, pp. 1–6.

[8] W. Mou and A. Kleiner, "Online Learning Terrain Classification for Adaptive Velocity Control," in *SSRR*, 2010, pp. 1–7.

[9] M. H. Bharati, J. J. Liu, and J. F. MacGregor, "Image texture analysis: Methods and comparisons," *Chemometrics and Intelligent Laboratory Systems*, vol. 72, no. 1, pp. 57–71, 2004.

[10] X. Li and Z. Tian, "Wavelet Energy Signature: Comparison and Analysis," pp. 474–480, 2006.

[11] S. Grigorescu, N. Petkov, and P. Kruizinga, "Comparison of texture features based on Gabor filters," *IEEE Transactions on Image Processing*, vol. 11, no. 10, pp. 1160–1167, 2002.

[12] S. Dash and U. R. Jena, "Texture classification using Steerable Pyramid based Laws' Masks," *Journal of Electrical Systems and Information Technology*, pp. 1–13, 2016.

[13] Y. Gao, C. Spiteri, M.-T. Pham, and S. Al-Milli, "A survey on recent object detection techniques useful for monocular vision-based planetary terrain classification," *Robotics and Autonomous Systems*, vol. 62, no. 2, pp. 151–167, 2014.

[14] S. Bartoszyk, P. Kasprzak, and D. Belter, "Terrain-aware motion planning for a walking robot," in *RoMoCo*, 2017, pp. 29–34.

[15] T. Homberger, M. Bjelonic, N. Kottege, and P. V. K. Borges, "Terrain-dependant Control of Hexapod Robots using Vision," in *International Symposium on Experimental Robotics*, 2016, pp. 92–102.

[16] R. B. Rusu, Z. C. Marton, N. Blodow, and M. Beetz, "Learning informative point classes for the acquisition of object model maps," *ICARCV*, pp. 643–650, 2008.

[17] R. B. Rusu, N. Blodow, and M. Beetz, "Fast Point Feature Histograms (FPFH) for 3D registration," *ICRA*, pp. 3212–3217, 2009.

[18] R. B. Rusu, G. Bradski, R. Thibaux, and J. Hsu, "Fast 3D recognition and pose using the viewpoint feature histogram," *IROS*, pp. 2155–2162, 2010.

[19] A. Aldoma, M. Vincze, N. Blodow, D. Gossow, S. Gedikli, R. B. Rusu, and G. Bradski, "CAD-model recognition and 6DOF pose estimation using 3D cues," in *ICCV*, 2011, pp. 585–592.

[20] S. Boris, E. Lin, J. A. Bagnell, J. Cole, N. Vandapel, and A. Stentz, "Improving Robot Navigation Through Self-Supervised Online Learning," *Journal of Field Robotics*, vol. 23, no. 11-12, pp. 1059–1075, 2006.

[21] Z. C. Marton, D. Pangercic, N. Blodow, J. Kleinehellefort, and M. Beetz, "General 3D modelling of novel objects from a single view," *IROS*, pp. 3700–3705, 2010.

[22] S. Lazebnik, C. Schmid, J. Ponce, S. Lazebnik, C. Schmid, and J. Ponce, "A sparse texture representation using local affine region," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1265–1278, 2010.

[23] D. M. Bradley, J. K. Chang, D. Silver, M. Powers, H. Herman, P. Rander, and A. Stentz, "Scene Understanding for a High-mobility Walking Robot," *IROS*, vol. 2015-Decem, pp. 1144–1151, 2015.

[24] M. Kragh, R. N. Jørgensen, and H. Pedersen, "Object Detection and Terrain Classification in Agricultural Fields Using 3D Lidar Data," in *ICVS*, vol. 9163, Copenhagen, 2015, pp. 188–197.

[25] C. A. Brooks and K. Iagnemma, "Self-supervised terrain classification for planetary surface exploration rovers," *Journal of Field Robotics (JFR)*, vol. 29, no. 3, pp. 445–468, 2012.

[26] C.-c. Chang and C.-j. Lin, "LIBSVM : A Library for Support Vector Machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, pp. 1–39, 2013.

[27] M. R. Heinen, P. M. Engel, and R. C. Pinto, "IGMN : An Incremental Gaussian Mixture Network that Learns Instantaneously from Data Flows," *Enia*, vol. ENIA, p. 12, 2011.

[28] R. C. Pinto and P. M. Engel, "A Fast Incremental Gaussian Mixture Model," *PLoS ONE*, vol. 10, no. 10, pp. e0139931+, 2015.

[29] P. Domingos and G. Hulten, "Mining high-speed data streams," in *Sixth ACM SIGKDD International Conference*, 2000, pp. 71–80.

[30] V. da Silva and A. T. Winck, "Video popularity prediction in data streams based on context-independent features," in *SAC*, 2017, pp. 95–100.

[31] J. Mrva and J. Faigl, "Tactile sensing with servo drives feedback only for blind hexapod walking robot," in *RoMoCo*, 2015, pp. 240–245.

[32] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

Online Incremental Learning of the Terrain Traversal Cost in Autonomous Exploration

Miloš Prágr   Petr Čížek   Jan Bayer   Jan Faigl

Abstract—In this paper, we address motion efficiency in autonomous robot exploration with multi-legged walking robots that can traverse rough terrains at the cost of lower efficiency and greater body vibration. We propose a robotic system for online and incremental learning of the terrain traversal cost that is immediately utilized to reason about next navigational goals in building spatial model of the robot surrounding. The traversal cost experienced by the robot is characterized by incrementally constructed Gaussian Processes using Bayesian Committee Machine. During the exploration, the robot builds the spatial terrain model, marks untraversable areas, and leverages the Gaussian Process predictive variance to decide whether to improve the spatial model or decrease the uncertainty of the terrain traversal cost. The feasibility of the proposed approach has been experimentally verified in a fully autonomous deployment with a hexapod walking robot.

I. INTRODUCTION

Multi-legged walking robots are capable of rough terrains traversal, either by leveraging detailed foothold position plans [2, 37], or reactively utilizing tactile information [4, 10]. On the other hand, the robots may suffer from poor energy efficiency [13] and low stability [18]; hence, they can benefit from traversal cost prediction of the observed terrains. In unknown environments, the robot may encounter previously unobserved terrain types, and therefore, it needs to explore and actively update its terrain traversal cost model to improve its performance as the perception is active by nature [1].

We propose to address autonomous robotic exploration as a problem to simultaneously create a spatial model of the unknown environment together with incremental learning of the traversal cost model. The spatial model is employed to reason about untraversable areas, but incrementally learned traversal cost characterizes the robot experience with its locomotion effectiveness over traversable terrains. Thus, the learned model is employed in the extrapolation of the traversal cost assessment to observed but not yet visited areas, to intentionally avoid hard-to-traverse areas.

The spatial frontier-based exploration [35] can be utilized to navigate the robot towards passable areas at the boundary of the explored space. However, there is not an easily distinguishable boundary in exploring some underlying phenomena such as the terrain traversal cost. Therefore, model confidence can
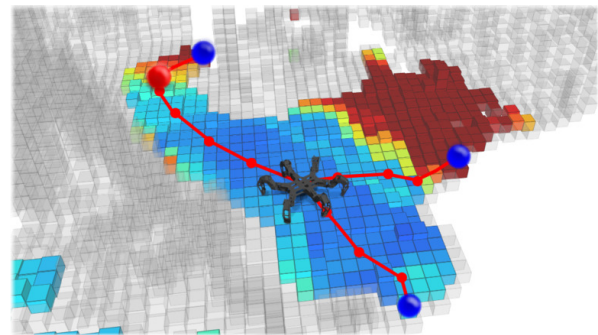
Fig. 1. Visualization of reasoning about possible navigational goals in the spatial frontier-based (blue spheres) exploration combined with building terrain traversal cost model (jet color coding). The robot selects to actively improve its traversal cost model for areas with low model confidence (red spheres). The model is instantly utilized in path planning, and thus the robot avoids areas that are believed to be hard-to-traverse.

be used to reason about active navigation towards areas with low model fidelity [29, 34]. An example of such reasoning within the proposed approach is visualized in Fig. 1.

The confidence of the terrain traversal cost model can be obtained using Gaussian Process (GP) regression, a nonparametric generalization of the linear regression which can extrapolate both the predictive mean and variance. The GP regression has been used in spatial exploration with a continuous spatial occupancy [20, 26] or terrain elevation models [32]. However, the GP regression suffers from cubic learning time complexity, and the Bayesian Committee Machine (BCM) with the GP regressors in frontier-based exploration [14] has been utilized to create spatial occupancy representation [33].

Motivated by recent advancements on the GPs and BCM, we propose to use the Robust BCM (RBCM) [8] for incremental construction of the terrain traversal cost model in the exploration of unknown environments. The robot thus alters navigation towards frontiers of the spatial exploration and areas with low confidence of the traversal cost model that is characterized by a high predictive variance; while simultaneously exploiting the learned model to avoid costly terrains.

In this paper, we describe the developed robotic system that represents an integrated framework with complete pipeline of sensing, model building, informed planning, and execution that has been experimentally verified in autonomous experiments. Regarding the existing work, we consider the main contributions of the presented approach as follows.

- Robotic system with active improvements of the terrain traversal cost model deployed in the autonomous exploration of the spatial model and the traversal cost model.
- Experimental validation of the proposed system in autonomous exploration with the hexapod walking robot.
- Deployment of online incremental learning of the underlying traversal cost model using the RBCM with GP regressor experts over the terrain feature descriptor space.
- Experimental evaluation of (fast) incremental learning approaches within the addressed terrain traversal cost modeling task.

The rest of the paper is structured as follows. Related approaches on terrain traversal cost model and characterization are overviewed in the following section. A brief description of the used RBCM is presented in Section III. The main parts of the proposed framework are described in Section IV and results on its experimental validation are reported in Section V. The paper is concluded in Section VI.

II. OVERVIEW OF THE EXISTING TERRAIN TRAVERSAL COST AND TERRAIN CHARACTERIZATION APPROACHES

Robots autonomously navigating in rough terrains must identify and avoid risks such as possible robot damage or energy wasting due to low efficiency. The risk and efficiency of the terrain traversal can be defined by characterizing remotely observed terrains or by examining the robot experience of the actual traversal. Observed geometrical and appearance properties of the perceived environment can be used for a remote characterization but the traversal cost defined as the level of risk and locomotion efficiency needs to be based on the robot traversal experience with the terrain.

Geometric properties such as height [13, 28], slope [6], or roughness [16] are directly connected to the viability of the terrain traversal. Multiple geometric properties can be used to detect unpassable areas and describe safe terrains, e.g., by a combined danger index in [28]. The appearance descriptors leverage the frequency domain or the colors of the observed areas. In [27], Gabor filters are used to describe overhead imagery, while voxel color information is directly utilized in [3], and color and reflectance vegetation indices are reviewed in [36]. Appearance and geometric descriptors may not only directly define the terrain traversal feasibility but can also be used to learn alternative terrain characterizations such as the terrain classification and robot experience.

Terrain classification is a task to assess the terrain into a set of discrete terrain classes based on human labeled terrain types [21] or to cluster unlabeled terrains [11]. Individual terrain types can carry information about the terrain traversability, e.g., an unpassable obstacle class [5]. Terrain classification can be based on geometrical and appearance properties of remotely observed terrains [16] but also on proprioceptive sensing [11]. A combination of the appearance and vibration terrain characterization is, for example, utilized in [19, 21].

Experience with the terrain traversal can be characterized as the observed difficulty of the robot with walking over the traversed terrain, and it can encode the tradeoff between various measures of the traversal efficiency. The Static Stability Margin [18] and the Dynamic Stability Margin [17] measure the stability of the multi-legged robot by observing its support polygon, i.e., the polygon defined by its footholds, and projection of the center of gravity. The concept of robot stability is also related to the vehicle vibration, which may decrease the perception accuracy, and eventually damage the robot construction. Alternatively, the terrain traversal experience can be encoded in a performance measure such as the cost of transport that is defined as the ratio between the consumed energy and velocity of the robot [15, 31].

The proposed framework (described in Section IV) is tailored to employ any continuous experience-based performance measures, but for our herein presented particular setup with hexapod walking robot, the terrain traversal risk is characterized as the experienced stability of the robot over a predefined time window. However, such a traversal risk is a particular instance for the system deployment, and thus its description is dedicated to Section V.

III. GAUSSIAN PROCESS REGRESSION

A brief overview of the Robust Bayesian Committee Machine (RBCM) inference mechanism is presented here together with a summary of the GP regression and BCM with GP regressor experts to make the paper more self-contained.

GP regression is a non-parametric generalization of the linear regression and for given noisily observed function $f(x)$

$$y = f(x) + \epsilon, \quad \epsilon \in \mathcal{N}(0, \sigma^2), \tag{1}$$

the GP is defined as a distribution over functions [25]

$$f(x) \sim \mathcal{GP}(m(x), K(x, x')) \tag{2}$$

characterized by its mean $m(x)$ and covariance $K(x, x')$ as

$$m(x) = E\left[f(x)\right], \tag{3}$$

$$K(x, x') = E\left[(f(x) - m(x))(f(x') - m(x'))\right], \tag{4}$$

for any pair of (x, x') out of the input space $\mathcal{X}$. Given the training data $(X, y) = \{X_i, y_i\}_{i=1}^n$ with the size n, the predictive equations of the latent values f_* for the test data X_* can be determined as

$$\begin{aligned} \mu(X_*) &= K_* \left[K + \sigma^2 I\right]^{-1} y, \\ (\sigma(X_*))^2 &= K_{**} - K_*^T \left[K + \sigma^2 I\right]^{-1} K_*, \end{aligned} \tag{5}$$

where the notation for $(\sigma(X_*))^2$ is abused to improve the readability and clarity of the equation, and K, K_*, and K_{**} are covariance matrices defined as

$$K = K(X, X), K_* = K(X, X_*), K_{**} = K(X_*, X_*). \tag{6}$$

The GP regression has cubic learning time complexity $\mathcal{O}(n^3)$, which limits its application in tasks such as the robotic exploration, and therefore, less demanding approach is desirable.

The BCM [30] is a product of experts approach that allows combining GP regressors learned on multiple datasets. Since each of the GP regressors can be constructed independently, the learning time complexity of the BCM with GP regressors
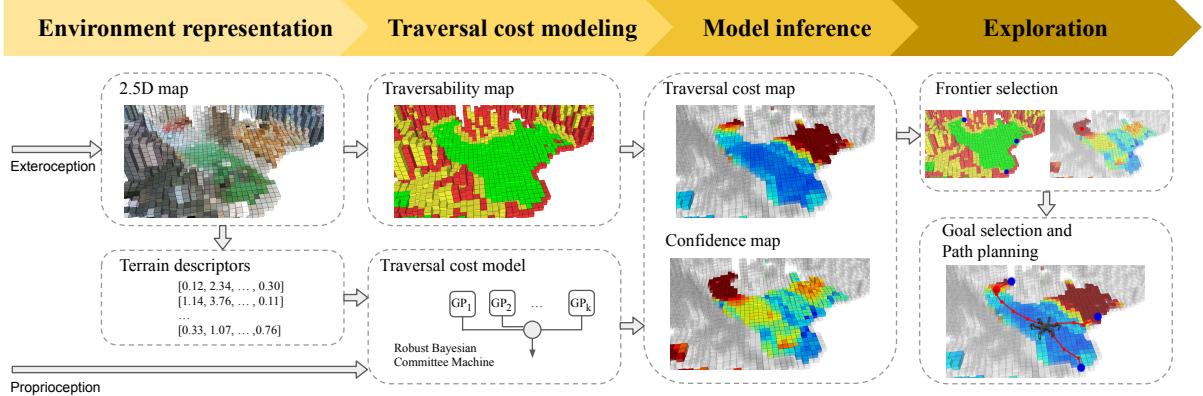
Fig. 2. Overview of the proposed system for online incremental learning of the terrain traversal cost in autonomous spatial exploration.

is $\mathcal{O}((n/k)^3 k)$, where k is the number of equally sized components of the size m, i.e., $n = km$. In practice, the size of a single component m is significantly smaller than n, and thus its processing time can be considered constant. Thus for a fixed m and $m \ll k$ we can consider $\mathcal{O}(m^3 k) \sim \mathcal{O}(k)$ and since k is proportional to n, the total complexity of k regressors can be approximately bounded by $\mathcal{O}(n)$, a notable improvement over the original $\mathcal{O}(n^3)$. The predictive equations of the BCM with k Gaussian regression experts can be defined as

$$\mu_{\text{BCM}}(X_*) = (\sigma_{\text{BCM}}(X_*))^2 \sum_{i=1}^{k} (\sigma_i(X_*))^{-2} \mu_i(X_*),$$

$$(\sigma_{\text{BCM}}(X_*))^{-2} = (1-k)(K_{**})^{-1} + \sum_{i=1}^{k} (\sigma_i(X_*))^{-2},$$

$$(7)$$

where $\mu_i(X_*)$ and $(\sigma_i(X_*))^2$ are the means and covariances of the individual experts. The BCM has been further improved as the Robust BCM (RBCM) in [8]. The RBCM weights the individual experts based on the predictive power of each expert at X_* and the RBCM predictive equations are

$$\mu_{\text{RBCM}}(X_*) = (\sigma_{\text{RBCM}}(X_*))^2 \sum_{i=1}^{k} \beta_i(X_*)(\sigma_i(X_*))^{-2} \mu_i(X_*),$$

$$(\sigma_{\text{RBCM}}(X_*))^{-2} =$$
$$= \left(1 - \sum_{i=1}^{k} \beta_i(X_*)\right)(K_{**})^{-1} + \sum_{i=1}^{k} \beta_i(X_*)(\sigma_i(X_*))^{-2},$$

$$(8)$$

where the weight β_i of the expert i at X_* is defined as

$$\beta_i(X_*) = 0.5 \left(\log(K_{**}) - \log((\sigma_i(X_*))^2)\right). \quad (9)$$

Thus, $\beta_i(X_*)$ is the difference in the differential entropy between the prior $p(f_*|x_*)$ and posterior $p_i(f_*|x_*, X^i)$, where X^i is the training dataset of the i-th expert.

IV. Autonomous Exploration with Online Incremental Terrain Traversal Cost Learning

In the addressed problem, we are motivated to build a fully autonomous system capable of operating in a priory unknown

environment and without prior knowledge about the traversal cost. We consider the proposed system within an autonomous exploration setup to simultaneously build a spatial model of the operational environment together with the traversal cost model that is learned incrementally to increase confidence in the cost estimation. Thus, during the exploration, the robot reasons how to improve the spatial and traversal cost models while it leverages on the experience accumulated in the incrementally learned traversal cost model in navigating towards spatial frontiers and avoiding costly terrains. The whole system consists of individual modules to build the spatial environment model, mark unpassable terrains, learn the terrain traversal cost model characterizing traversable terrains (continuously utilized in the determination of the next exploration goal), navigation to the selected goal, and locomotion control.

The overall system architecture can be divided into four main parts that are depicted in Fig. 2. Exteroceptive signals are processed in the environment representation to localize the robot and build a map of the robot surroundings together with extracting terrain shape and appearance feature descriptors that are further utilized in traversal cost model inference. The traversal cost modeling includes the incremental learning of the traversal cost model using the robot proprioceptive experience coupled with the terrain descriptors. Besides, non-traversable parts of the environment are labeled as areas with infinite traversal cost based on the geometric features of the created terrain elevation map to avoid unnecessary model-based traversability assessment of unpassable areas.

The traversal cost model is employed in model inference to build a cost map of reachable areas of the environment together with the confidence of the estimated cost that is utilized in the selection of the next exploration goal towards which the robot is navigated. The system continuously gathers new measurements and updates the current navigational goal until no exploration goal is determined. A detailed description of the individual parts of the system follows.

A. Environment Representation

The environment is represented as the colored 2.5D elevation grid map denoted $\mathcal{M}_{2.5\text{D}}$ (see Fig. 3a), which utilizes an

Fig. 3. Colored elevation grid map $\mathcal{M}_{2.5D}$ (a); traversability grid map $\mathcal{M}_{trv}$ (b); traversal cost grid map $\mathcal{M}_{cost}$ (c); and the respective confidence of the traversal cost model (d). Note, $\mathcal{M}_{cost}$ characterizes only the traversable areas.
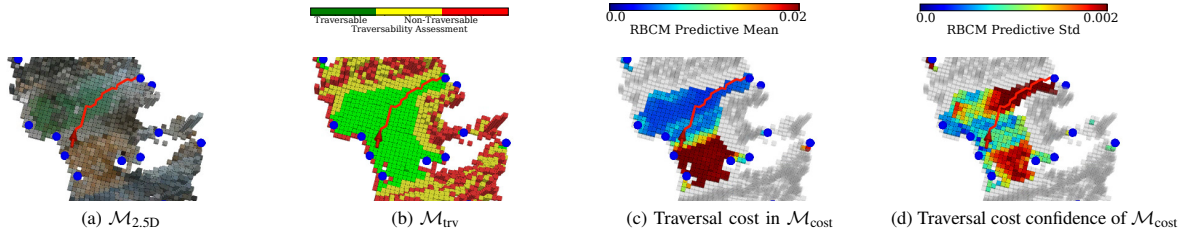
underlying quadtree data structure. Each cell $\nu \in \mathcal{M}_{2.5D}$ stores elevation and RGB color information, and it is further characterized with the five-dimensional geometric and appearance terrain feature descriptor $desc(\nu)$ that is a modification of the terrain descriptor used in [23]. The used geometric part of the descriptor, which is designed to distinguish the unstructured, linear, and planar shape [16] of the terrain, is defined as

$$s_1 = \frac{\lambda_1}{\lambda_3}, \quad s_2 = \frac{\lambda_2 - \lambda_1}{\lambda_3}, \quad s_3 = \frac{\lambda_3 - \lambda_2}{\lambda_3}, \quad (10)$$

where $\lambda_1 < \lambda_2 < \lambda_3$ are the eigenvalues of the covariance matrix of the elevation and spatial values in the spatial δ_{desc}-neighborhood of the cell ν. The residual sum of the squares feature utilized in [16, 23] is relaxed, and the two-dimensional appearance part of the descriptor is the δ_{desc}-neighborhood channel means of the ab channels of the Lab color space. For further information on the performance of individual descriptor parts and their combinations, we kindly refer the reader to [23].

B. Traversal Cost Modeling

The main role of the traversal cost modeling is to incrementally learn the traversal cost model based on the terrain feature descriptors. The particular cost value captures the real robot experience with the particular terrain type and it is measured by proprioceptive sensing. The model consists of the RBCM with GP regressor experts learned to allow inferring the traversal cost observed by the robot from the geometric and appearance description of the terrain. Thus, the RBCM is augmented with the terrain descriptors paired with the traversal cost experienced over the said terrain. Since the RBCM with a single expert would behave similarly to the GP; therefore, the maximal size of expert m_{max} is specified as a tradeoff between the computational requirements and achieved precision of the model. The RBCM experts are constructed incrementally, and a new expert is allocated every m_{max} observations. In particular, $m_{max} = 50$ is selected so that a new expert is allocated approx. once per minute because of the robot locomotion speed. Each expert thus consists of spatially neighboring terrains. In addition, we further specify the minimal number of observations m_{min} to consider the particular expert in the inference, because small experts may negatively spoil it with their high confidence.

In addition to the model learning, unpassable terrain areas are marked at this stage, and the $\mathcal{M}_{2.5D}$ grid map is transformed into the traversability grid map $\mathcal{M}_{trv}$. Inspired by [37],

we determine untraversable cells using step height defined as the maximum elevation difference of the neighboring cells. Thus, based on the motion capabilities of the particular robot, we specify the maximum h_{max} for traversable cells, and thus cells with the step height above h_{max} are considered untraversable. Besides, considering the embodiment of the robot, we further mark cells in the $\delta_{impassable}$-radius neighborhood of such an untraversable cell also as untraversable, see Fig. 3b.

C. Model Inference

In the model inference part of the framework, the traversal cost grid map $\mathcal{M}_{cost}$ is created combining $\mathcal{M}_{2.5D}$, $\mathcal{M}_{trv}$, and the current learned terrain traversal cost model. Each cell $\nu' \in \mathcal{M}_{cost}$ characterizes the inferred traversal cost accompanied with the traversal cost model confidence over the traversable regions represented by $\mathcal{M}_{trv}$, see Figs. 3c and 3d, respectively. In general, the resolution of $\mathcal{M}_{cost}$ can differ from $\mathcal{M}_{2.5D}$ as the resolution affects the level of details achieved in spatial exploration, model exploration, but also path planning, and the most suitable resolution of the individual maps depends on the sensor resolution and the size and step length of the robotic platform. It might be necessary to resample maps using terrain descriptors of $\mathcal{M}_{2.5D}$ and model inference for each grid cell that is traversable according to $\mathcal{M}_{trv}$. Thus, for each traversable grid cell $\nu' \in \mathcal{M}_{cost}$, the closest grid cell $\nu \in \mathcal{M}_{2.5D}$ is determined and its terrain descriptor denoted $desc(\nu)$ is inferred. The model inference is employed for each traversable cell $\nu' \in \mathcal{M}_{cost}$ to estimate the cost using the traversal cost model prediction mean for the descriptor $desc(\nu)$

$$\mu(\nu') = \mu_{RBCM}(desc(\nu)), \quad (11)$$

and the model confidence is determined as the square root of the traversal cost prediction variance for $desc(\nu)$

$$\sigma(\nu') = \sigma_{RBCM}(desc(\nu)), \quad (12)$$

where high $\sigma(\nu')$ signifies low model confidence.

D. Exploration

The exploration module selects the next navigational goal location towards which the robot navigates. Different strategies to tradeoff the spatial exploration with the model improvement can be designed, but the proposed approach combines spatial frontiers and traversal cost model exploration. The employed strategy greedily improves the traversal cost by navigating to

---

Algorithm 1: Autonomous exploration with online incremental terrain traversal cost learning

Input: $\delta_{\text{desc}}, h_{\text{max}}, \delta_{\text{impassable}}, m_{\text{min}}, r_{\text{spatial}}, \delta_{\text{score}}, \lambda_{score}$ – Parameters of the terrain characterizing descriptors, traversability map, terrain traversal cost map and navigational goals.

```
1  repeat
2      M_2.5D ← getSpatialModel(δ_desc)                              // Using exteroception, e.g., RGB-D data
3      M_trv ← getTraversabilityMap(M_2.5D, h_max, δ_impassable)           // Mark untraversable areas using [37]
4      RBCM ← getTravelCostModel(m_min)                // Get experts, each with at least m_min observationss
5      M_cost ← inferModel(RBCM, M_2.5D, M_trv)
6      ν_spatial ← getBestSpatialGoal(M_2.5D, M_cost, M_trv, r_spatial)              // Using cost (11) and (14)
7      (ν_cost, score(ν_cost)) ← getBestCostGoal(M_cost, M_trv, δ_score)                  // Using (12) and (15)
8      ν* ← selectNextGoal(ν_spatial, ν_cost, score(ν_cost), scoreThreshold)      // Select the next goal using (16)
9      path(ν_r, ν*) ← findPath(M_cost, ν_r, ν*)    // Path from the current robot position ν_r to the goal ν*
10     setNavigationWaypoints(path(ν_r, ν*))            // Set the path as the next navigational waypoints
11 until ν* is ∅
```

getBestSpatialGoal($\mathcal{M}_{2.5D}$, $\mathcal{M}_{cost}$, $\mathcal{M}_{trv}$, $r_{spatial}$):
```
1  F_spatial ← clusterFrontiers(M_2.5D)                                   // Cluster frontiers using (13)
2  N_spatial ← assignReachableCells(F_spatial, M_cost, M_trv, r_spatial)       // Place reachable frontiers on M_cost
3  return Dijkstra(N_spatial, M_cost)                           // Select the cheapest goal to reach using (14)
```

getBestCostGoal($\mathcal{M}_{cost}$, $\mathcal{M}_{trv}$, δ_{score}):
```
1  forall reachable ν ∈ M_cost do
2      Σ ← {}                                             // Compute score for all reachable cells
3      forall ν' ∈ M_cost where ‖(ν, ν')‖ < δ_score do
4          Σ ← Σ ∪ σ(ν')              // Collect uncertainties for all cells in neighborhood using (12)
5      score(ν) ← median(Σ)                               // Compute the score using (15)
6  return (argmax(score), max(score))            // Select the reachable cell with the highest score
```

terrains that are considered unknown. If the observed terrains are sufficiently known, the robot explores the spatial frontiers.

Spatial goal locations are determined as means of clustered frontiers (representatives), where each cluster is a single connected component of the selected frontier cells. The number representative n_r of a single component is determined as [9]

$$n_r = 1 + \left\lfloor \frac{f}{D} + 0.5 \right\rfloor, \tag{13}$$

where f is the current number of frontier cells and D is the sensor range (in the number of grid cells). The set of frontier cells $\mathcal{N}_{spatial} \subset \mathcal{M}_{\text{cost}}$ is created by assigning a reachable, and thus traversable cell $\nu \in \mathcal{M}_{\text{cost}}$ that is incident with an unexplored cell. Moreover, frontier cells closer than $r_{\text{spatial}} = 0.4$ m to the current robot position ν_r are also ignored to avoid navigating to goals that the robot cannot observe en-route. The best spatial goal $\nu_{spatial}$ is selected as the representative with the lowest cost to be reached from the current robot position ν_r. The cheapest to reach spatial exploration goal is determined using Dijkstra's algorithm with the traversing cost $c(\nu_i, \nu_j)$ between two neighborhood cells $\nu_i, \nu_j \in \mathcal{M}_{\text{cost}}$ based on the cost prediction (11) and Euclidean distance $\|(\nu_i, \nu_j)\|$ between the centers of the corresponding grid cells ν_i and ν_j as

$$c(\nu_i, \nu_j) = \|(\nu_i, \nu_i)\| \left(\mu(\nu_i) + \mu(\nu_j)\right)/2. \tag{14}$$

Dijkstra's algorithm is preferred since we need to determine the cost to reach all representatives, but the closest is selected.

The goal locations for the traversal cost model are grid cells $\nu \in \mathcal{M}_{\text{cost}}$ with a high model uncertainty that is considered as the $score(\nu)$ over δ_{score} spatial neighborhood to characterize the level of details in the model exploration. The value of $score(\nu)$ for a cell $\nu \in \mathcal{M}_{\text{cost}}$ is defined as the median of the traversal cost predictive standard deviation (12) of the neighboring cells

$$score(\nu_i) = \text{median}\{\sigma(\nu_j) \mid \|(\nu_i, \nu_j)\| < \delta_{\text{score}}\}. \tag{15}$$

The cell with the highest score is selected as the next navigational goal locations for exploring the traversal cost model.

Then, a particular exploration strategy is employed, which in our case is a preference of the model learning. Thus, the robot prefers to explore unknown terrains (if any has been observed) and the next navigational goal is the center of the grid cell ν^* selected using the decision rule

$$\nu^* = \begin{cases} \nu_{cost} & \text{if } score(\nu_{cost}) > \lambda_{score}, \\ \nu_{spatial} & \text{if a reachable spatial goal location exists}, \\ \emptyset & \text{otherwise}, \end{cases}$$
$$\tag{16}$$

where λ_{score} is the threshold model confidence to do not consider further improvement of the traversal cost model. The particular value λ_{score} needs to be set to fit the traversal costs range and variance observed by the utilized robotic platform.

Finally, once the navigational goal ν^* is determined, the cheapest path is computed by Dijkstra's algorithm using (14)

and such a path is used to navigate the robot towards the goal. Since the robot collects new information during its navigation in the environment, and thus it improves its spatial and traversal cost model continuously, it is desirable to perform the decision-making at a high rate to quickly exploit new knowledge about the environment. Both the spatial model and traversal cost model are independently updated in separate execution loops defined by the maximal processing speed of the localization and model learning. The exploration loop needs to build (update) the traversability map $\mathcal{M}_{trv}$ and infer a new traversal cost map $\mathcal{M}_{cost}$ based on the spatial map $\mathcal{M}_{2.5D}$. Besides, the updated RBCM-based traversal cost model is employed to determine new goals and plan the paths.

A summary of the exploration loop with the individual parts of the model usage is depicted in Algorithm 1 together with the list of parameters that are specified for the particular experimental deployment in the following section.

V. EXPERIMENTAL RESULTS AND DISCUSSION

The proposed system for autonomous exploration with online and incremental learning of the terrain traversal cost has been experimentally validated in two scenarios. First, we compare the performance of the RBCM with GP regressor experts to pure GP regression on a dataset captured by the utilized robot. Second, we deploy the system in a fully autonomous exploration to demonstrate the online model learning and its benefit for avoiding costly terrains experienced by the robot.
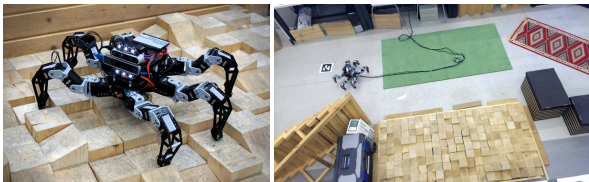
Fig. 4. The utilized hexapod walking robot and the laboratory test track.

The system is deployed on a small hexapod walking robot operating on a laboratory test track with rough terrains (see Fig. 4). The developed system consists of several modules and implementations overviewed in Table I. The substantial part of the proposed approach is the traversability cost determined from the proprioceptive measurements. Based on the review of existing work (see Section II), we chose stability based traversal cost computed as the variance of the roll angle of the sliding window spanning the robot gait cycle duration of $10\,s$. We found out that a high roll variance indicates the robot cannot find firm footholds, which decreases its speed, risks damage to its body, and also hampers the perception accuracy.

The robot has been deployed in the laboratory test track with six selected terrains denoted *flat*, *grass*, *carpet*, *cubes*, *ramp*, and *stairs* that display specific interaction properties when traversed by the hexapod walking robot as follows.

- PVC flooring (*flat*) represents an easy to traverse terrain.
- Turf-like carpet (*grass*) and red carpet (*carpet*) are soft terrains with faint response to the foot contact.

- Wooden blocks (*cubes*) with different height and slope with the base of $10 \times 10\,cm$ represent a harder to traverse rough terrain mock-up with the overall size of $2.3 \times 1.2\,m$.
- Hard to traverse ramp (*ramp*) and wooden stairs (*stairs*) with $4\,cm$ steps inducing vibrations due to slippage.

TABLE I
INDIVIDUAL PARTS OF THE DEVELOP ROBOTIC SYSTEM

Part / Module	Used Setup / Utilized Implementation
Robot	Hexapod walking robot with six legs each with three actuated joints. The robot dimensions are about $45 \times 40\,cm$ when standing in a default configuration, and we set $h_{max} = 0.2\,m$ and $\delta_{impassable} = 0.25\,m$.
Locomotion control	We employ the available approach [10] with the mean walking velocity of the robot around $0.05\,ms^{-1}$. The robot employs the follow the carrot algorithm with $20\,cm$ distance threshold for the path following.
Exteroception, Proprioception	Intel RealSense D435 (RGB-D imagery, 640×480 at $15\,Hz$), Intel RealSense T265 (localization $200\,Hz$).
Computational resources	Intel Core i7-8650u CPU with $16\,GB$ RAM, Ubuntu 18.04, and implementation in ROS Melodic [24].
Environment map	$\mathcal{M}_{cost}$ grid map with the grid cell size $10\,cm$, which roughly corresponds to the robot step length for a single gait cycle, and thus we chose $\delta_{desc} = 0.2\,m$ and $\delta_{score} = 0.5\,m$ to prefer larger patches of terrains.
Traversability cost	The variance of the roll angle θ_{roll} over the sliding window $10\,s$ long.

A. Comparison of the RBCM and Pure GP-based Regression

In this experiment, we compare the performance of the herein utilized RBCM with GP regressor experts with the pure GP-based regression using real dataset collected by the used robot in the laboratory test track. We are specifically focused on the evaluation of the prediction abilities and computational requirements in the online incremental learning setup, and therefore, we consider six learning setups: the RBCM with Exponential, Matérn $3/2$, Matérn $5/2$, and RBF kernels; Incremental Gaussian Mixture Network (IGMN) [22], which is a representative of a broader set of fast incremental approaches; and pure GP learned using all the available observations, which serves as the baseline approach that is however expected to be computationally demanding. The individual GP experts and the pure GP model are learned using the GPy framework [12], and each of the individual experts is optimized using the limited memory Broyden-Fletcher-Goldfarb-Shanno with boundaries (L-BFGS-B) [7] limited to 200 steps.

All the methods are evaluated on the dataset containing 292 data points of terrain descriptors paired with the corresponding stability based traversal cost which ranges in $[0.005, 0.050]$, with a considerable variance over the wooden stairs. The dataset comprises two human operated runs over the laboratory test track. Each model is incrementally learned on the dataset; i.e., a single observed data point is added to the model at each learning step. Besides, at each step, the models are used to predict the traversal cost for the whole dataset, and the prediction is compared to the set of the collected measurements. Since the selected traversal cost encodes the robot experience with traversal of terrains, it cannot be obtained without the

(a) The learning time observed in the RBCM experimental verification.

(b) The inference time observed in the RBCM experimental verification.

(c) RMSE on the full dataset observed in the RBCM experimental verification.

(d) The model confidence represented as the model prediction standard deviation. Note, GP uses the alternative scale.
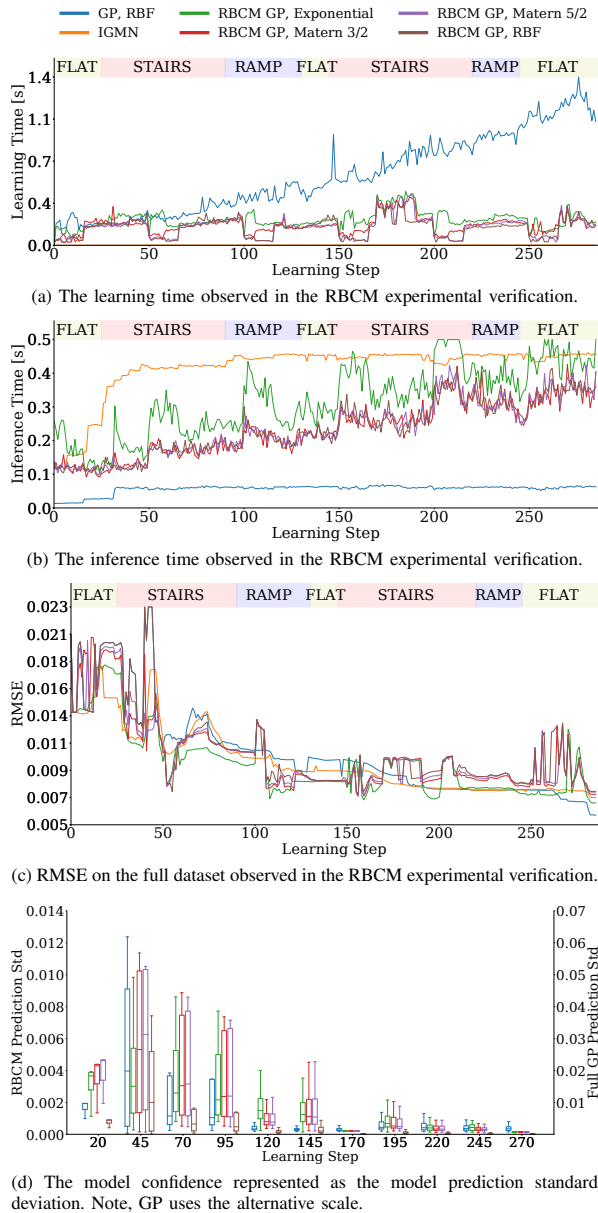
Fig. 5. Results of the comparison of the RBCM and Pure GP-based regression.

measurement noise caused by the robot motion, and thus the set is considered to represent the traversal cost ground truth. We also report the time to learn and predict the traversability costs of the whole dataset. We have processed the dataset in 100 learning-and-prediction trials, and report the mean value to profile the proposed system and mitigate the effects of OS scheduling, garbage collection, etc. The computational times to learn the individual models, the inference times, and to predict means and standard deviations are reported in Fig. 5.

The results indicate that the fastest learning method is the IGMN, which fuses a new observation with $\mathcal{O}(1)$, although

the theoretical time complexity of the RBCM is the same for the constant m_{max}, as only one expert of the limited size needs to be retrained. The GP has the fusing complexity $\mathcal{O}(n^3)$ since it retrains the whole model at each time step. The inference times reported in Fig. 5b slowly increase for the RBCM as the number of experts gradually increases. The evolution of the Root Mean Squared Error (RMSE) is shown in Fig. 5c. The peaks in the RBCM correspond to the influence of the newly inserted small experts with a low number of observations, which are overly confident due to a few observations. Thus, it is desirable to set the expert minimum size m_{min} in the inference. At the end of the testing trail, the IGMN provided similar results to most of the RBCM variants, but the GP learned the best representation of the terrain costs; however, the RBCM models outperform the GP before the second traverse of the experimental mock-up when the robot revisits already visited areas. Even though the IGMN performs similarly to the RBCM, its main drawback is that it does not predict model confidence. The results in Fig. 5d show that the predictive standard deviation of the RBCM models is considerably lower than that of the GP. Both the RBCM and GP exhibit similar behavior when the predictive standard deviation is lowered by adding observations, and the predictive standard deviation for unobserved terrains is higher than the predictive standard deviation for rough terrains with varied traversal cost measurement.

In conclusion, the RBCM with GP regressor experts provides similar performance to the GP regressor regarding both performance indicators, the RMSE, and the model confidence observed on the verification dataset. Moreover, the RBCM learning is less computationally demanding and can satisfy the real-time requirements of the online deployment with the real walking robot. On the other hand, the RBCM inference time is higher, resulting in a tradeoff between the complexity of learning and inference, with a preference on the learning speed in the herein presented deployment. The best performing RBCM with the size $m_{max} = 50$, $m_{min} = 25$ and the exponential kernel is utilized in the autonomous exploration deployment reported in the following section.

B. Autonomous Exploration

In the experimental deployment of the proposed system in fully autonomous exploration, the laboratory test track has been surrounded by boards and boxes to guarantee the exploration is finished within a reasonable time. The considered terrains for this experiment are the *flat* ground, the wooden *cubes*, the artificial *grass*, and red *carpet*. During the experiment, the robot has been placed in the arena and requested to explore the whole area and build the terrain traversal cost model. The verification of the system performance is made by observing the robot behavior and profiling the implementation.

During the operation, the proposed system fully exploits the computational resources. The update rate and CPU usage for each individual part of the system are reported in Table II. The majority of the resources has been consumed by the implemented exploration node that is responsible for the

(a) Situation after partial creation of the first model component. The robot chooses to explore the traversal cost over the wooden *cubes*.

(b) The robot revisits the wooden *cubes* to improve its model.

(c) The robot has explored the traversal cost model over all terrains it has observed so far, and thus the system switched to the spatial exploration.

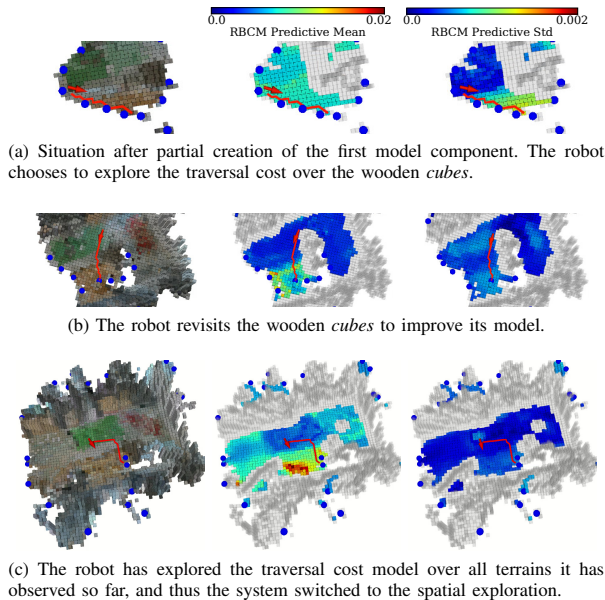Fig. 6. Evaluation of the traversable terrain at partial time instants of the autonomous exploration. From left: the robot position on the test track, predicted traversal cost, and confidence of the traversal cost model.

construction of $\mathcal{M}_{cost}$ and exploration. At the beginning of the experimental trail, the exploration node took $1.1\,\mathrm{s}$ to plan a new path for a map of 219 traversable cells, in the middle of the experiment it took $10.0\,\mathrm{s}$ for 793 traversable cells, and at the end $23.2\,\mathrm{s}$ for a map with 910 cells. The speed of the $\mathcal{M}_{2.5D}$ update fluctuated based on the number of concurrently updated nodes. However, the performance of the exteroception, proprioception, model learning, and locomotion control have been stable.

TABLE II
PERFORMANCE OF THE AUTONOMOUS EXPLORATION

Process	CPU usage*	Update rate
Exteroception	36%	15 Hz
Proprioception - Traversal cost calculation	54%	200 Hz
$\mathcal{M}_{2.5D}$ and $\mathcal{M}_{trv}$ construction and terrain descriptors calculation	98%	1–30 Hz[+]
Model learning	61%	1 Hz
$\mathcal{M}_{cost}$ inference and exploration	450%	0.05–1 Hz
Locomotion control	19%	–

* Maximal CPU usage 800% (4 cores with Hyper-threading)
[+] Depending on the size of the map update

The exploration deployment has been performed several times with similar behavior. The hexapod walking robot has been deployed using either the *adaptive* motion gait [10], which is designed to traverse rough terrains, or with the *regular* gait, which is faster but less capable on rough terrains. The value of λ_{score} has been set to 0.001 and 0.0005 for the *adaptive* and *regular* gait, respectively.

The behavior of the robot in the selected showcase situations is documented in Fig. 6, and the complete evolution of the spatial and traversal cost models over one trial is presented in the accompanying video. Altogether 11 experts with $m_{max} = 50$ have been created in the total during this trial. At the beginning of each experiment, the robot is located on the *flat* ground and it is allowed to freely explore the environment. As the robot did not yet traverse any terrain, the terrain traversal cost and the model confidence are both uniform, and the robot chooses to go towards the nearby spatial goal. After the first model component is created, the robot typically explores some area with a low model confidence (represented by a high predictive standard deviation), e.g., the wooden *cubes* or the centrally located artificial *grass*, see Fig. 6a. Occasionally, the robot briefly samples traversal cost over a terrain type, and continues on to explore another terrain that is unknown. In such a case, the robot may return to the first terrain to further improve its model, see Fig. 6b. If the robot has observed all the readily available terrains, it continues with spatial exploration, see Fig. 6c. Although it is not possible to provide ground truth for the learned model, because it depends on the particular trial, the robot has always identified the *flat* ground, artificial *grass*, and red *carpet* as an easy to traverse, and the rough wooden *cubes* as hard to traverse.

Based on the experimental deployment, we can conclude that the robotic system presented in this paper is capable of exploiting both the spatial and traversal cost model exploration, and selects the one that suits its currently accumulated knowledge with the preference towards traversal cost model exploration. The system is also capable of making informed decisions and intentionally avoid hard-to-traverse areas.

VI. CONCLUSION

In this paper, we present a robotic system for spatial exploration that is combined with the exploration of underlying traversal cost model over traversable terrains that is enabled by employing the Robust Bayesian Committee Machine (RBCM) with GP regressor experts, which learning part is less computationally demanding than the pure GP regressor, and thus it is more suitable for online decision-making. The additional advantage of using the RBCM (e.g., in comparison to the IGMN) to create the terrain traversal cost model incrementally is that it provides both the predictive mean and variance for the observed terrains, and thus allows the robot to explore areas of low model fidelity. The proposed approach has been deployed in the developed robotic system and verified in a fully autonomous exploration with the hexapod walking robot. The reported experimental results support the robot is capable of exploiting the spatial knowledge and make informed decisions and intentionally avoid hard-to-traverse terrains during the exploration. We aim to extend the approach by incorporating a combined probabilistic representation of the terrain traversal feasibility and terrain traversal cost, and also consider intended robot maneuvers, because the terrain traversability differs based on the particularly performed robot maneuver.

REFERENCES

[1] R. Bajcsy. Active Perception. *Technical Reports (CIS)*, March 1988.

[2] S. Bartoszyk, P. Kasprzak, and D. Belter. Terrain-aware motion planning for a walking robot. In *RoMoCo*, pages 29–34. IEEE, 2017. doi: 10.1109/RoMoCo.2017. 8003889.

[3] D. Belter, J. Wietrzykowski, and P. Skrzypczynski. Employing Natural Terrain Semantics in Motion Planning for a Multi-Legged Robot. *Journal of Intelligent & Robotic Systems*, pages 1–21, 2018. doi: 10.1007/ s10846-018-0865-x.

[4] G. Bledt, P. M. Wensing, S. Ingersoll, and S. Kim. Contact Model Fusion for Event-Based Locomotion in Unstructured Terrains. In *ICRA*, pages 1–8, 2018. doi: 10.1109/ICRA.2018.8460904.

[5] D. M. Bradley, J. K. Chang, D. Silver, M. Powers, H. Herman, P. Rander, and A. Stentz. Scene Understanding for a High-mobility Walking Robot. In *IROS*, pages 1144–1151. IEEE, 2015. doi: 10.1109/IROS.2015. 7353514.

[6] M. Brunner, B. Brüggemann, and D. Schulz. Rough Terrain Motion Planning for Actuated, Tracked Robots. In *ICAART*, pages 40–61. Springer, 2013. doi: 10.1007/ 978-3-662-44440-5_3.

[7] R. Byrd, P. Lu, J. Nocedal, and C. Zhu. A Limited Memory Algorithm for Bound Constrained Optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, September 1995. doi: 10.1137/0916069.

[8] M. Deisenroth and J. W. Ng. Distributed Gaussian Processes. *arXiv:1502.02843*, July 2015.

[9] J. Faigl, M. Kulich, and M. Přeučil. Goal Assignment using Distance Cost in Multi-Robot Exploration. In *IROS*, pages 3741–3746. IEEE, 2012. doi: 10.1109/ IROS.2012.6385660.

[10] Jan Faigl and Petr Čížek. Adaptive Locomotion Control of Hexapod Walking Robot for Traversing Rough Terrains with Position Feedback Only. *Robotics and Autonomous Systems*, 116:136–147, 2019. doi: 10.1016/ j.robot.2019.03.008.

[11] P. Giguere and G. Dudek. Clustering Sensor Data for Terrain Identification using a Windowless Algorithm. In *Robotics: Science and Systems IV*. Robotics: Science and Systems Foundation, June 2008. doi: 10.15607/RSS. 2008.IV.004.

[12] GPy. GPy: A gaussian process framework in python. http://github.com/SheffieldML/GPy, since 2012.

[13] T. Homberger, M. Bjelonic, N. Kottege, and P. V. K. Borges. Terrain-dependant Control of Hexapod Robots using Vision. In *ISER*, pages 92–102. Springer, 2016. doi: 10.1007/978-3-319-50115-4_9.

[14] M. G. Jadidi, J. V. Miro, and G. Dissanayake. Gaussian processes autonomous mapping and exploration for range-sensing mobile robots. *Autonomous Robots*, 42(2):273–290, February 2018. doi: 10.1007/

[15] N. Kottege, C. Parkinson, P. Moghadam, A. Elfes, and S. P. N. Singh. Energetics-informed hexapod gait transitions across terrains. In *ICRA*, volume 2015-June, pages 5140–5147. IEEE, 2015. doi: 10.1109/ ICRA.2015.7139915. 10.1109/ICRA.2015.7139915 978-1-4799-6923-4.

[16] M. Kragh, R. N Jørgensen, and H. Pedersen. Object Detection and Terrain Classification in Agricultural Fields Using 3D Lidar Data. In *ICVS*, volume 9163, pages 188–197. Springer, 2015. doi: 10.1007/978-3-319-20904-3_ 18.

[17] B. Lin and S. Song. Dynamic modeling, stability and energy efficiency of a quadrupedal walking machine. In *ICRA*, pages 657–670. IEEE, May 1993. doi: 10.1002/ rob.8104.

[18] R. B. McGhee and A. A. Frank. On the stability properties of quadruped creeping gaits. *Mathematical Biosciences*, 3:331–351, August 1968. doi: 10.1016/ 0025-5564(68)90090-4.

[19] W. Mou and A. Kleiner. Online Learning Terrain Classification for Adaptive Velocity Control. In *SSRR*, pages 1–7. IEEE, 2010. doi: 10.1109/SSRR.2010.5981563.

[20] S. O'Callaghan, F. T. Ramos, and H. Durrant-Whyte. Contextual occupancy maps using Gaussian processes. In *ICRA*, pages 1054–1060. IEEE, May 2009. doi: 10.1109/ROBOT.2009.5152754.

[21] K. Otsu, M. Ono, T. J. Fuchs, I. Baldwin, and T. Kubota. Autonomous Terrain Classification with Co- and Self-Training Approach. *RAL*, 1(2):1–6, 2016. doi: 10.1109/ LRA.2016.2525040.

[22] R. Pinto and P. Engel. A Fast Incremental Gaussian Mixture Model. *PLOS*, page e0141942, 2015. doi: 10.1371/journal.pone.0139931.

[23] M. Prágr, P. Čížek, and J. Faigl. Cost of Transport Estimation for Legged Robot Based on Terrain Features Inference from Aerial Scan. In *IROS*, pages 1745–1750. IEEE, 2018. doi: 10.1109/IROS.2018.8593374.

[24] Morgan Quigley, Ken Conley, Brian P. Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y. Ng. ROS: an open-source Robot Operating System. In *ICRA Workshop on Open Source Software*, 2009.

[25] C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, Cambridge, Mass, 2006.

[26] A. V. Ruiz and C. Olariu. A general algorithm for exploration with Gaussian processes in complex, unknown environments. In *ICRA*, pages 3388–3393, Seattle, WA, USA, May 2015. IEEE. doi: 10.1109/ICRA. 2015.7139667.

[27] B. Sofman, E. Lin, J. A. Bagnell, J. Cole, N. Vandapel, and A. Stentz. Improving Robot Navigation Through Self-Supervised Online Learning. *Journal of Field Robotics*, 23(11-12):1059–1075, 2006. doi: 10.1002/rob.

20169.

[28] A. Stelzer, H. Hirschmüller, and M. Görner. Stereo-vision-based navigation of a six-legged walking robot in unknown rough terrain. *International Journal of Robotics Research*, 31(4):381–402, 2012. doi: 10.1177/0278364911435161.

[29] S. B. Thrun. Exploration and model building in mobile robot domains. In *ICNN*, pages 175–180 vol.1. IEEE, March 1993. doi: 10.1109/ICNN.1993.298552.

[30] V. Tresp. A Bayesian Committee Machine. *Neural Computation*, 12(11):2719–2741, November 2000. doi: 10.1162/089976600300014908.

[31] V. A. Tucker. The Energetic Cost of Moving About: walking and running are extremely inefficient forms of locomotion. Much greater efficiency is achieved by birds, fish and bicyclists. *American Scientist*, 63(4):413–419, 1975.

[32] S. Vasudevan, F. Ramos, E. Nettleton, H. Durrant-Whyte, and A. Blair. Gaussian Process modeling of large scale terrain. In *ICRA*, pages 1047–1053. IEEE, May 2009. doi: 10.1002/rob.20309.

[33] J. Wang and B. Englot. Fast, accurate gaussian process occupancy maps via test-data octrees and nested Bayesian fusion. In *ICRA*, pages 1003–1010. IEEE, May 2016. doi: 10.1109/ICRA.2016.7487232.

[34] P. Whaite and F. P. Ferrie. Autonomous exploration: driven by uncertainty. *Transactions on Pattern Analysis and Machine Intelligence*, 19(3):193–205, March 1997. doi: 10.1109/34.584097.

[35] B. Yamauchi. A frontier-based approach for autonomous exploration. In *CIRA*, pages 146–151. IEEE, July 1997. doi: 10.1109/CIRA.1997.613851.

[36] C. Ünsalam and K. Boyer. Linearized vegetation indices based on a formal statistical framework. *Transactions on Geoscience and Remote Sensing*, 42(7):1575–1585, 2004. doi: 10.1109/TGRS.2004.826787.

[37] P. Čížek, D. Masri, and J. Faigl. Foothold Placement Planning with a Hexapod Crawling Robot. In *IROS*, pages 4096–4101. IEEE, 2017. doi: 10.1109/IROS.2017.8206267.

Check for updates

Autonomous robotic exploration with simultaneous environment and traversability models learning

Miloš Prágr*, Jan Bayer and Jan Faigl

Computational Robotics Laboratory, Faculty of Electrical Engineering, Czech Technical University in Prague, Prague, Czechia

In this study, we address generalized autonomous mobile robot exploration of unknown environments where a robotic agent learns a traversability model and builds a spatial model of the environment. The agent can benefit from the model learned online in distinguishing what terrains are easy to traverse and which should be avoided. The proposed solution enables the learning of multiple traversability models, each associated with a particular locomotion gait, a walking pattern of a multi-legged walking robot. We propose to address the simultaneous learning of the environment and traversability models by a decoupled approach. Thus, navigation waypoints are generated using the current spatial and traversability models to gain the information necessary to improve the particular model during the robot's motion in the environment. From the set of possible waypoints, the decision on where to navigate next is made based on the solution of the generalized traveling salesman problem that allows taking into account a planning horizon longer than a single myopic decision. The proposed approach has been verified in simulated scenarios and experimental deployments with a real hexapod walking robot with two locomotion gaits, suitable for different terrains. Based on the achieved results, the proposed method exploits the online learned traversability models and further supports the selection of the most appropriate locomotion gait for the particular terrain types.

KEYWORDS

mobile robot exploration, active learning, traversability, multi-legged robot, locomotion gait

1 Introduction

The presented online terrain learning approach is motivated by long-term missions where autonomous robots would improve their operational performance in navigating *a priori* unknown environments. Some difficult to traverse terrains, such as large rocks, can be identified as obstacles using an observed geometric model of the environment. However, areas which appear flat and thus easy to traverse may, in practice, be hard to traverse due to their terra-mechanical properties, as experienced by NASA's Mars Rover Spirit stuck in soft sand (Brown and Webster, 2010). In the presented approach,
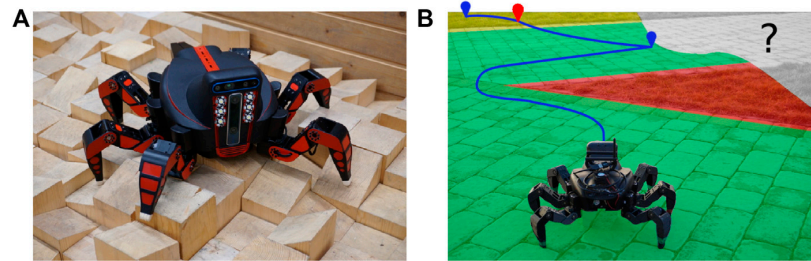
FIGURE 1
(A) Hexapod walking robot (courtesy of Forouhar et al. (2021)) **(B)** and its deployment using the proposed approach. The visualized planned path is to visit determined exploration goals for the spatial (in blue) and traversal cost models (in red). The spatial exploration goals are located close to the boundary of the already explored part of the environment. The traversal cost exploration goals correspond to sites where the terrain traversal cost model can be improved. Since the cost model is already partially learned, the red-tinted turf is known to be hard to traverse, and thus the robot prefers the green-tinted pavement, which is relatively easy to traverse. The yellow-tinted terrain is yet to be experienced by the robot and thus carries the terrain learning goal indicated by the red waypoint. The not-yet-observed area is gray.

individual terra-mechanical properties are assumed to be partially unknown, and we learn a black box model to assess the traversability in a particular environment from the terrain appearance (Prágr et al., 2018). Since the scope of the functional relation between the terrain appearance and traversability might be limited to a particular environment, we advocate that on long-term deployments and exploration missions, the terrain models are learned online incrementally (Prágr et al., 2019b) as a part of the mission (Prágr et al., 2019a). Hence, we focus on the exploration of the environment and its terra-mechanical properties represented as the traversal costs that characterize the difficulty of traversing the individual terrains, as visualized in Figure 1. In particular, we consider multi-legged walking robots that can traverse various terrains with different traversal costs (also depending on the particular locomotion gait used), which provide a representative case for demonstrating the benefits of traversability assessment learned online. Compared to the previous work, the presented approach addresses the different locomotion gaits of the robot and distinguishes individual terrain-gait traversal cost models. In addition, the proposed exploration strategy provides a non-myopic (Zlot and Stentz, 2006) solution that takes into account both the spatial exploration and learning of the traversal cost models.

In the proposed approach, the impassable parts of the explored environment are determined by the geometric models using a grid-based elevation map (Bayer and Faigl, 2019). The individual terrain-gait traversal cost models are near-to-far predictors that infer the time to traverse over the traversable areas from their appearance and are learned using the robot's previous experience accrued when traversing similar-appearing terrains using a particular gait. The traversal cost models comprise Gaussian process (GP) regressors (Rasmussen and Williams, 2006), which predict the traversal costs from the terrain appearance, and growing neural gas

(GNG) (Fritzke, 1994) terrain type clustering schemes used to identify similar-appearing terrains. The geometric and traversal cost models are incrementally constructed while exploring the mission environment. The geometric model is continually built from the robot's exteroception, whereas each traversal cost model accumulates the costs experienced by the robot when moving using the respective locomotion gait. During the deployment, each model continually provides a set of exploration goals to be visited to learn (improve) the model. For several possible goal locations, the exploration strategy is to determine a sequence of the navigational goals to be visited that is addressed as a solution of the *Generalized Traveling Salesman Problem* (GTSP) (Noon, 1988) to provide a non-myopic solution considering the so-called TSP distance cost (Faigl and Kulich, 2013).

The remainder of the article is organized as follows. In Section 2, we present an overview of the related approaches in mobile robot exploration and traversability assessment. Section 3 formally defines the studied problem of mobile robot exploration with *a priori* unknown terrain traversal cost assessment. The proposed exploration with online traversal cost learning is presented in Section 4. Section 5 reports on the performed experimental results in simulations and real-world experimental deployments with a multi-legged robot controlled by two motion gaits. In Section 6, we discuss the strong points and limitations of the proposed approach. Section 7 concludes the study.

2 State of the art

This section presents an overview of works related to the proposed approach. First, we focus on the traversability assessment approaches. Then we survey mobile robot exploration and environment modeling.

50

2.1 Mobile robot traversability

Two main questions emerge when reasoning about robot traversability over terrains. First, can the terrain be safely traversed, or should it be avoided? Second, if the terrain is passable, how does it compare to other terrains, i.e., is it easier and safer to traverse? Note that for the sake of clarity, we further denote the binary (`true`/`false`) traversability, which determines whether an area is an impassable obstacle or passable terrain, as terrain passability. In contrast, the relative comparison of the traversal difficulty over passable terrains is denoted as assessing the traversal cost. The term traversability is used to describe the notion in general, including both the passability and traversal cost. A review of mobile robot traversability assessment methods can be found in Papadakis (2013), and an overview of learning-based methods for ground robot navigation is in Guastella and Muscato (2021). Hence, we focus on works relevant to how traversability is approached in this study.

The passability discrimination can be directly incorporated in mapping in the form of occupancy cell grids (Moravec and Elfes, 1985), Gaussian mixtures (O'Meadhra et al., 2019), GP models (O'Callaghan et al., 2009), or Hilbert maps (Ramos and Ott, 2016). The distinction of terrain passability can be understood as an instance of terrain classification, where terrains are assigned individual classes, and each class carries presumed terra-mechanical properties. For example, some classes can be considered hard-to-traverse vegetation or obstacles (Bradley et al., 2015). In addition to terrain classification, terrains can be assigned continuous values describing some observed terrain property such as roughness (Krüsi et al., 2016; Belter et al., 2019), slope (Stelzer et al., 2012), or step height (Homberger et al., 2016; Wermelinger et al., 2016). For continuous measures, passability can be based on thresholding the value, as in Stelzer et al. (2012), where the passability is determined by individually thresholding terrain slope, roughness, and step height. Moreover, classes may correspond to a particular robot configuration, such as in Haddeler et al. (2020), where the authors classify terrains into modes of wheeled-legged locomotion.

In instances where the terra-mechanical properties are unknown and thus terrains' appearance and geometry features are not sufficient to determine their traversability, the traversability can be based on the robot's prior experience with similar terrains. The experience-based measures can be derived from the robot proprioception and described using stability (McGhee and Frank, 1968; Lin and Song, 1993), slippage (Gonzalez and Iagnemma, 2018), vibrations (Bekhti and Kobayashi, 2016), velocity, or energy consumption (Kottege et al., 2015). The experience-based approaches describe the traversal cost only over passable terrains since the traversal is needed to acquire the robot experience. An exception worth mentioning is haptic sensing to determine obstacle

passability (Baleia et al., 2015), which, however, still relies on the direct interaction of the robot with the terrain.

Since the experience-based approaches use on-location robot experience, they are difficult to use directly in path planning where it is necessary to evaluate terrain traversability from a distance using only exteroceptive measurements. Near-to-far approaches pair traversability indicators that can be observed only near the robot (such as proprioception or dense short-range measurements) with terrain appearance and geometry that can be observed from farther distances and thus learn to predict traversability from the long-range measurements. Sofman et al. (2006) incrementally learned the relation between dense laser-based features characterizing ground unit traversability and overhead features that can be used to assess traversability from aerial images, whereas Bekhti and Kobayashi (2016) learned to predict vibration-based traversability from terrain texture. Quann et al. (2020) proposed an energy traversal cost regressor considering both terrain position and appearance. In addition, Mayuku et al. (2021) proposed a self-supervised labeling approach for a near-to-far scenario, where vibration-based traversal cost is inferred from image data, and the self-supervised data gathering is based on identified terrain classes.

Following the approaches in the literature, we assume that terrain is rigid, and it is possible to distinguish passable terrain and non-traversable obstacles from the terrain geometry using a step height similar to Stelzer et al. (2012), or Wermelinger et al. (2016). Hence, this study focuses on modeling the traversal cost over the determined passable terrains. Moreover, we are motivated by the online cost assessment in mobile robot exploration, where the computational requirements are crucial. Therefore, we avoid high fidelity models, which besides being costly to compute also rely on plan execution with high precision (such as deterministic foothold placement), which might not be available in practice. The traversal cost is thus learned as a black box near-to-far model that uses terrain appearance to predict the time to traverse over terrains. Since the scope of the relation between the terrain appearance and traversability might be limited to a particular environment, we incrementally learn the cost predictor by sampling the robot's experience with traversing individual terrains. Similar to the classification in Belter et al. (2019), a color histogram is selected as the terrain appearance descriptor because it is simple to compute and the histograms are sufficiently descriptive to capture multi-colored terrains. Furthermore, we consider locomotion gaits of the employed hexapod walking robot that are suitable for different terrains. Thus, the passable terrain is a terrain traversable by at least one gait, and obstacles are terrain parts that none of the gaits can traverse. We propose a decoupled approach that predicts the traversal cost for each gait independently, and the robot then selects the most cost-efficient gait for each terrain.

Regarding the existing methods, the proposed approach is closest to Haddeler et al. (2020), where modes of the wheeled-

legged robot are switched. In addition, the proposed approach is also close to the self-supervised, near-to-far traversability-learning approach proposed by Mayuku et al. (2021). In that regard, the primary contribution of the proposed approach is the integration of active traversability learning in mobile robot exploration, where the robot plans a non-myopic path to improve both the spatial and traversal cost models learned online during the deployment.

2.2 Mobile robot exploration and environment modeling

Mobile robot exploration is an active perception problem that concerns behaviors where the robot seeks to build a model of *a priori* unknown environment. The exploration entails the robot seeking areas that are in some capacity unknown to construct a map of the environment. The exploration thus inherently combines localization, navigation, and planning (Schultz et al., 1999) to decide where the robot should go next. Steering the robot navigation to not-yet-observed areas yields frontier-based exploration (Yamauchi, 1997), where the frontiers represent boundaries between the observed traversable area and the unknown space represented on an occupancy grid (Moravec and Elfes, 1985). Recently, in the octree-based environment model, frontiers are represented as mesh faces with few neighbors (Azpúrna et al., 2021).

Bourgault et al. (2002) and Makarenko et al. (2002) exploit the probabilistic representation on such an occupancy evidence grid and navigate to maximize the approximated occupancy information gain. Charrow et al. (2015) proposed to use Cauchy–Schwarz quadratic mutual information to speed up the information gain computation. In addition, approaches that rely on non-grid-based representation for navigation, such as meshes and topological maps, may retain cell or voxel grids to quantify the information gain (Dang et al., 2020).

In addition to mapping, robots also build models of environment-underlying phenomena that can be temperature models (Luo and Sycara, 2018) or spread of gas (Rhodes et al., 2020). The environment phenomenon can be considered spatial, and the goal is thus to learn the mapping from the position in the environment to the value of the phenomenon. Furthermore, a spatiotemporal model can be considered (Ma et al., 2018) that would require repeatedly visiting particular areas to build the temporal model, which might be needed for changing environments (Krajník et al., 2017).

Spatial-based modeling can be considered as informative path planning (Singh et al., 2007), where the goal is to find the most informative path through the environment (Hollinger and Sukhatme, 2014) subject to a particular constraint such as the robot energy budget (Binney and Sukhatme, 2012). Informative path planning approaches can be broadly divided into myopic and non-myopic methods. The myopic methods are greedy and plan only with regard to the next goal, whereas non-myopic methods plan with a longer horizon. For example, in the context of frontier-based mobile robot exploration, seeking the closest frontier is myopic, contrary to path planning to visit all the representatives of the frontiers that is non-myopic (Faigl et al., 2012).

Like seeking frontiers in spatial exploration, the explorer learning an underlying model must actively locate sites to sample novel information. Hence, GP regressors (Rasmussen and Williams, 2006) are particularly suited for active learning because it is relatively straightforward to identify uncertain regions where the model should be improved. GP prediction uncertainty is characterized by the differential entropy of the predicted normal distribution, leading to the characterization of information gained by observing individual areas. However, in practice, directly computing the information gained by possible observations is not feasible due to the number of possible actions, especially for a long planning horizon. Hence, various approximations and sampling strategies have been proposed.

Pasolli and Melgani (2011) proposed to either directly seek the most uncertain samples signified by the highest prediction variance or to select areas that are the most remote in the feature space given the GP hyper-parameters. In Viseras et al. (2019), the robot selects paths with high average entropy per sampling to tradeoff informativeness and the number of samplings. Martin and Corke (2014) proposed to set the mean function of a GP traversal cost regressors to zero, thus motivating a robot to traverse unknown areas where the predictions are close to the zero mean. The *GP Upper Confidence Bound* (GP-UCB) (Srinivas et al., 2010) is an exploration–exploitation method that combines seeking the most uncertain areas with improving the model around the highest value. It can be used when the learner is interested in finding extreme values of the modeled phenomenon, such as temperature (Luo and Sycara, 2018; Shi et al., 2020). In addition, a depth-first variant of the *Monte Carlo Tree Search* (MCTS) to select anytime informative paths can be employed to consider both differential entropy and upper confidence bound to model sampling informativeness (Guerrero et al., 2021).

Karolj et al. (2020) computed a path to the closest spatial frontier that visits all local sampling locations for a magnetism model by solving the *Traveling Salesman Problem* (TSP) over the respective goal locations. In localization in mapping, Ossenkopf et al. (2019) note that occupancy information gained at an unknown location holds little value and thus weight the occupancy gains by a pose uncertainty (Vallvé and Andrade-Cetto, 2015). Hence, the explorer must address how to combine the occupancy and pose uncertainties. In Bourgault et al. (2002) and Stachniss et al. (2005), the total exploration utility is a linear combination of the occupancy uncertainty and the robot localization uncertainty represented using the differential entropy based on its position distribution. In Carrillo et al.

(2018), it is argued that combining Shannon's discrete and differential entropies is neither practical nor sound because the differential entropy is neither invariant under a change of variable nor dimensionally correct. Therefore, both quantities may differ significantly in value. Consequently, Carrillo et al. (2018) proposed to use the localization uncertainty to weigh the Rényi entropy (Rényi, 1961) of the occupancy grid.

Based on the literature review on exploration approaches, we propose to generalize the previous work (Prágr et al., 2019a) toward a non-myopic approach. The therein proposed method combines active learning of traversal cost over terrains with spatial exploration using a greedy approach. The approximated spatial information gains and cost models are derived from Shannon's discrete and differential entropies, respectively. Considering the reasoning of Carrillo et al. (2018), we avoid a direct combination of these two values in this study. In addition, we aim to build a modular system that supports the learning of models that range from the spatial map and cost predictors used in this study to temperature and pollution models. Hence, instead of creating a combined information gain utility function using the Rényi entropy, which is suitable for the combination of a map and robot's localization model used by Carrillo et al. (2018), we elect to use a policy that combines the spatial exploration and cost learning goals (and goals reported by any additional model), similarly to the approach proposed by Karolj et al. (2020).

However, unlike the therein-built magnetism model, a spatial GP, we assume that the terrain traversal cost correlates with the terrain appearance. Therefore, the GP regressor infers the cost from the terrain feature descriptors instead of the terrain location. Consequently, rather than terrains nearby, sampling the cost to traverse an unknown terrain primarily affects the predictions over similarly appearing terrains close in the feature space. The affected terrains are determined using a terrain clustering scheme. Incremental growing neural gas (IGNG) (Prudent and Ennaji, 2005) is used to continually construct the terrain class structure, in which each class is assigned traversal cost and sampling reward (information gain) based on the GP's predictions. As a result, we model the computation of the goal visit sequence as an instance of the *Generalized TSP* (GTSP) (Noon, 1988) (also called the Set TSP), which is a variant of the TSP where nodes are grouped into mutually exclusive and exhaustive sets. The problem is then to visit each set instead of visiting each node. In the context of the proposed exploration approach, the individual nodes correspond to possible sampling locations, and the sets are either terrain classes extracted from the cost prediction model or places where the robot can observe areas unknown to the spatial model.

The problem of mobile robot exploration with traversal cost learning is defined in the next section, whereas the strengths and weak points of the proposed approach are further discussed in Section 6.

FIGURE 2
Footprint around the robot position covers the cells with potential multi-legged walking robot footholds.

3 Problem specification

The addressed exploration using an autonomous hexapod walking robot combines spatial exploration with active learning of terrain traversal cost models. The environment is modeled as a 2D grid $\mathbb{W} \subset \mathbb{R}^2$ with cells $v \in \mathbb{W}$ with size d_v corresponding to the size of the robot foothold. The position of the robot p^{robot} is discretized as v^{robot} within the grid that is at the center of the robot's circular footprint with radius r_{robot} covering all the potential robot's footholds, as shown in Figure 2. Any path ψ can be decomposed to a sequence of neighboring cells as follows:

$$\psi = (v_1, v_2, \ldots, v_n),$$
$$\text{s.t.}$$
$$\forall i \in 1, \ldots, n: \ \pi(v_i) = 1, \quad (1)$$
$$\forall i \in 1, \ldots, n-1: \ v_{i+1} \in 8\text{nb}(v_i),$$

where n is the number of cells in the respective sequence, the function $8\,\text{nb}(v)$ lists the cells in the 8-neighborhood of v, and $\pi(v) = 1$ indicates that the cell v is passable. In addition, the robot can use a discrete set of walking gaits $\mathbb{G}$, and it is assumed that the gait changes occur instantaneously at the particular grid cells $v \in \mathbb{W}$.

The robot desires to move through the environment as efficiently as possible with respect to (w.r.t.) the cost C. Therefore, it moves along the cheapest path between v and v'.

$$\psi^*(v, v') = \text{argmin}_{\psi \in \Psi(v, v')} C(\psi), \quad (2)$$

where $\Psi(v, v')$ is the space of all paths from v to v'. The cost $C(\psi)$ of traversing ψ represents a generic path cost such as time to traverse or expected consumed energy; without the loss of generality, the time to traverse is the cost of choice in this study. It is assumed that the cost is additive, thus permitting to combine the costs of two consecutive path segments ψ_a and ψ_b into the cost of the combined path $\psi_a \oplus \psi_b$ as follows:

$$C(\psi_a \oplus \psi_b) = C(\psi_a) + C(\psi_b), \qquad (3)$$

where $\oplus$ denotes the concatenation of the paths. The cost of a path is decomposed to the sequence of costs to traverse from passable cell v_a to its neighbor v_b.

$$C(\psi) = \sum_{i=1}^{n-1} \|(v_i, v_{i+1})\| c(v_i, v_{i+1}), \qquad (4)$$

where $\|(v_a, v_b)\|$ is the Euclidean distance between the cells (i.e., either d_v or $\sqrt{2}d_v$), and $c(v_a, v_b)$ is the per-meter cost of traversing from v_a to v_b.

In the spatial exploration, the robot builds the geometry model $\mathcal{P}$, which provides the cell passability assessment $\pi(v)$. It is assumed that the geometry is sufficient to distinguish the passable areas; hence, the passability model $\mathcal{P}$ is constructed directly from the continually streamed exteroceptive measurements (observed point clouds z^{pcd}).

3.1 Traversal cost modeling

The traversal cost is assumed to be too complex to be assessed only from the terrain geometry. In this study, the task is to learn a traversal cost predictor $\mathcal{C}$ that models the cost as a function of terrain appearance. The cost assessments are used in path planning w.r.t. (4). In addition, the cost model is also responsible for selecting the gaits suitable for the particular terrains traversed by the robot. Since the robot position is abstracted as the center of its circular footprint, the predictor's per-meter-cost predictions are conservative estimates that take into account all the cells on the footprint. The cost predictor is learned online during the exploration from the robot experience, which comprises the cost z^c experienced by the robot when traversing terrain described by the terrain appearance descriptor ta using gait g.

The learned model is compared to the uninformed baseline that represents a robot that only explores the spatial map and does not learn the cost models and thus uses the optimistic flat cost model.

$$\hat{c}(v_a, v_b) = \frac{1}{v_{\max}}, \qquad (5)$$

where $v_{\max}$ is the maximum robot velocity over all $g \in \mathbb{G}$. Notice that, in planning, the particular value of $v_{\max}$ is not relevant as long as it is positive because it only scales the total cost, thus not affecting the planning decisions. The baseline selects the gaits reactively, using the fast gait capable of reaching $v_{\max}$ by default and switching to slower yet rough-terrain-capable gaits when the robot gets stuck on the traversed terrain.

The proposed approach is evaluated in model scenarios as follows. First, the robot is set to explore the environment $\mathbb{W}$, and

it incrementally learns the model $\mathcal{C}$. Then the learned and baseline models are used in navigating the robot between a set of benchmark coordinates in $\mathbb{W}$ and the total cost C experienced by the robot (i.e., the time needed to move between the coordinates) using the particular model is considered to be the benchmark value.

4 Proposed system for active terrain learning in exploration

In this section, we describe the proposed system for active terrain learning and exploration, which is overviewed in Figure 3. During the exploration, which yields the spatial geometric passability model $\mathcal{P}$, the goal of the robot is also to learn the traversal cost model $\mathcal{C}$. The geometric passability model $\mathcal{P}$ describes the shape of the environment and thus areas passable by the robot. The traversal cost model is decomposed into the set of models $\mathcal{C} = \mathcal{C}^{\mathbb{G}} = \{\mathcal{C}^g\}_{g \in \mathbb{G}}$, where each traversal cost model $\mathcal{C}^g$ predicts the costs associated with traversing the passable terrain using the gait $g \in \mathbb{G}$. The respective cost predictors are Gaussian process (GP) regressors (Rasmussen and Williams, 2006), which use terrain appearance to infer the robot-experienced traversal cost accrued during the deployment. Each GP is coupled with the incremental growing neural gas (IGNG) (Prudent and Ennaji, 2005) that clusters similarly appearing terrains and hence identifies terrain types not yet visited by the robot. The exploration problem is modeled as an open-ended instance of the generalized traveling salesman problem (GTSP) (Noon, 1988), a variant of the TSP where the vertices are organized in disjoint sets, and each set is visited once. In this study, each set corresponds to an exploration or learning goal (a set of sampling sites) yielded by the spatial or cost model.

In the rest of the section, we describe the exploration process. The symbols used in the description are listed in Table 1. First, we show how the GTSP is used to find the exploration path. Then we show the geometric environment model in detail and the related passability model $\mathcal{P}$, the traversal cost models $\mathcal{C}^g$, and their use to find the exploration goals.

4.1 Exploration

The robot explores the passability model $\mathcal{P}$ and learns the traversal cost models $\mathcal{C}^{\mathbb{G}}$ by visiting the exploration $\Gamma_{\mathcal{P}}$ and cost learning $\Gamma_{\mathcal{C}}^{\mathbb{G}}$ goals, which are continually yielded by the respective models. Each goal $\gamma \in \Gamma_{\mathcal{P}} \cup \Gamma_{\mathcal{C}}^{\mathbb{G}}$ is associated with a set of sites (cells) $\gamma = \{v_i\}_{i=0}^{|\gamma|}$ where the robot can improve its models by sampling the respective goal. The robot needs to visit one of the corresponding locations to sample the goal. Geometric model goals $\gamma \in \Gamma_{\mathcal{P}}$ are located at singular sites $\gamma = $

FIGURE 3
An overview of the proposed exploration system. The robot uses the RGB-D data to build the color elevation model of the environment in which it identifies the passable areas (Algorithm 2). The terrain appearance stored in the model is paired with the costs experienced by the robot to learn the traversal cost models for the individual locomotion gaits (Algorithm 5 and Algorithm 6). The cost predictions for the individual gaits and the terrain passability are used to plan the exploration path in a TSP sequence (Algorithm 1) over every goal reported by the geometric and cost models. The robot navigates to the first goal in the sequence (Algorithm 4).

$\{\nu\}$, where the robot can improve the spatial model by observing new areas. Each traversal cost model goal $\gamma \in \Gamma_{\mathcal{C}}{}^{\mathbb{G}}$, where $\Gamma_{\mathcal{C}}{}^{\mathbb{G}} = \cup_{g \in \mathbb{G}} \Gamma_{\mathcal{C}}{}^g$, is associated with a set of sites $\gamma = \{\nu_i\}_{i=0}^{|\gamma|}$ at which the robot can improve the model by experiencing novel gait-terrain costs. The areas covered by the individual goals in a given cost model are designed to be disjoint. Thus, sampling the traversal cost model at a site corresponding to the goal ${}^2\gamma_{\mathcal{C}}{}^g \in \Gamma_{\mathcal{C}}{}^g$ provides no, or severely limited, information regarding the traversal cost model at a site corresponding to a different goal ${}^1\gamma_{\mathcal{C}}{}^g \neq {}^1\gamma_{\mathcal{C}}{}^g$. On the other hand, the passability and traversal cost models are considered independent. Sampling at one particular site might improve both models since the robot can observe previously unseen areas while experiencing untraversed terrain. However, two cost models cannot be improved at once since the robot can only experience the cost for the currently used gait.

Given the current robot position ν_t^{robot} and models $\mathcal{P}_t$ and $\mathcal{C}_t^{\mathbb{G}}$ at any time t during the exploration, the robot selects a shortest exploration path $\psi_E(p_t^{\text{robot}}, \mathcal{P}_t, \mathcal{C}_t^{\mathbb{G}})$ that visits at least one site corresponding to each goal. The path planning is modeled as an

instance of the GTSP, where vertices (sites) are organized in disjoint sets (goals), and each set is visited exactly once. The distance matrix D describes the costs of paths between the individual sites, including the distances between the current robot position and the goal sites.

$$D(\nu, \nu') = \hat{C}(\psi^*(\nu, \nu')). \tag{6}$$

A total of two transforms are applied to the distance matrix D to create an open instance of the GTSP. First, the robot does not need to return to its current position after exploring the environment. Hence, the problem is transformed by setting the cost to reach the current robot position from any goal as zero $\forall \gamma \in \Gamma_{\mathcal{P}} \cup \Gamma_{\mathcal{C}}^{\mathbb{G}}, \forall_\nu \in \gamma$: $D(\nu_\gamma, \nu^{\text{robot}}) = 0$. Second, we apply the Noon–Bean transformation (Noon and Bean, 1993) to transform an instance of the GTSP into an instance of the TSP. The open instances of the transformed TSP are solved by the LKH solver (Helsgaun, 2000), a heuristic solver with asymptotic time complexity bounded by $\mathcal{O}(m^{2.2})$, where m is the number of vertices, which has been found sufficient for updates with tens of goal sites. The solver returns the sequence

Appendix B.3 – Prágr *et al.*: "Autonomous Robotic Exploration with Simultaneous Environment and Traversability Models Learning" [C3], referenced on page 14.

TABLE 1 Used symbols.

Description	Symbol	Description	Symbol
World grid map model	$\mathbb{W}$	Grid map cell	ν
Grid map cell size	d_ν	Current robot position	ν^{robot}
Robot footprint radius	r_{robot}	Cell ν passability	$\pi(\nu)$
Path	ψ	Optimal path	ψ^*
Walking robot gait	g	Robot gait set	$\mathbb{G}$
Cost (time to traverse)	C	Per-meter cost	c
Geometric passability model	$\mathcal{P}$	Cost model	$\mathcal{C}$
Measured cost	z^c	Maximum robot velocity	v_{max}
Colored elevation grid map	$\mathcal{M}_{2.5D}$	Robot sensor range	r_{sensor}
Terrain appearance desciptor	ta	Descriptor radius	r_{hist}
Spatial clustering radius	c_{radius}	Cluster min cells	$c_{\text{min cells}}$
Cost model, all gaits	$\mathcal{C}^{\mathbb{G}}$	Cost model, particular gait	$\mathcal{C}^g$
Cost prediction, all gaits	$\hat{c}$	Cost prediction, particular gait	$\hat{c}_{\mathcal{C}^g}$
Distance transform per-meter loss	c_{loss}	Cost measurement variance	σ_{sense}^2
Cost measurement filter initial variance	σ_0^2		
GP regressor	$\mathcal{R}$	GP learning set	$\mathcal{L}$
GP prediction mean	$\hat{\mu}_c$	GP prediction variance	$\hat{\sigma}_c^{\;2}$
Prediction uncertainty/GP entropy	H	High cost in cost transform	c_{high}
Min learning set size	$n_{\mathcal{L}}^{\text{min}}$	GP model noise variance	$\sigma_\epsilon^{\;2}$
Exponential kernel length scale	l	Exponential kernel output variance	σ_s
Maximum allowed cost	c_{max}		
Terrain class model	$\mathcal{T}$	Terrain class	T
Approximated cost information gain	I_C	Terrain class uncertainty threshold	H_C^{GT}
Min GT terrain type size	m_T	Sampling lattice	S
Sampling lattice point	p_S	Sampling lattice size	d_S
Goal set	Γ	Goal	γ
Passability goal set	$\Gamma_{\mathcal{P}}$	Cost goal set, all gaits	$\Gamma_C^{\mathbb{G}}$
Cost goal set, particular gait	Γ_C^g	TSP distance matrix	D
Current exploration goal	ν_E^*	Current exploration path	ψ_E
Enforced sampling gait	g^{enforced}	Gait sampling duration	Δt_{sample}
IGNG structure	Ω	IGNG measurement	x
IGNG neuron set	Ω_{neurons}	IGNG connection set	$\Omega_{\text{connections}}$
IGNG neuron	ω	IGNG adaptation threshold	σ^{IGNG}
IGNG winner warp rate	ϵ_1^{IGNG}	IGNG neighbor warp rate	$\epsilon_{\text{nb}}^{\text{IGNG}}$
IGNG neuron mature age	$a_{\text{mature}}^{\text{IGNG}}$	IGNG connection maximum age	$a_{\text{max}}^{\text{IGNG}}$
Terrain type erosion steps	$n_{\text{erode}}^{\text{steps}}$	Terrain type dilation steps	$n_{\text{dilate}}^{\text{steps}}$
Terrain type dilation size	$n_{\text{dilate}}^{\text{size}}$		

of sites (ν^{robot}, ν_0, ν_1, ... , ν_n) to be visited through the environment, see Figure 4A, where n is the number of goals and each site ν_i corresponds to a different goal. The robot navigates toward the first site of the sequence and its current exploration goal ν_E^* becomes $\nu_E^* = \nu_0$, see an example of the path in Figure 4B.

The plan is recomputed on-demand either when there is a change in the goal set or as a result of reaching the current goal. Moreover, upon reaching a cost model goal, the robot switches to the model's respective gait g^{enforced} and is forced to move forward for Δt_{sample} (or until an obstacle is reached) to sample the traversal cost over the terrain. The exploration ends when

FIGURE 4

An example of a planned exploration path. **(A)** Global path over the sequence of goals determined by the TSP solver; **(B)** the local path to the first goal.
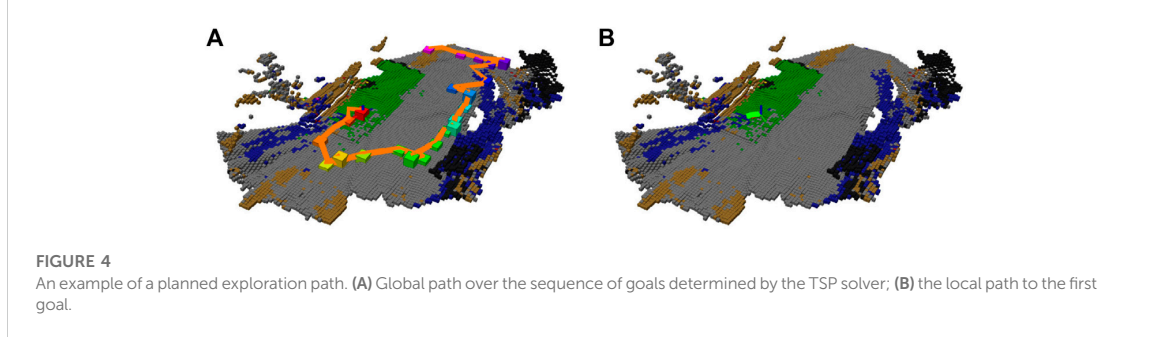
every model reports zero goals. The exploration process is summarized in Algorithm 1.

Algorithm 1. Exploration.

4.2 Environment geometry & passability model

The grid environment $\mathbb{W}$ is represented by the colored elevation grid map $\mathcal{M}_{2.5D}$ with the cell size d_v. The grid map is built online during the exploration according to Algorithm 2 using the robot's range measurements and RGB camera images. The elevation at each cell $v \in \mathcal{M}_{2.5D}$ is obtained by fusing the localized range measurements z_i^{pcd} into the grid map using an one dimensional Kalman filter, as described in Fankhauser et al. (2014) or Bayer and Faigl (2020). The localization of the robot, and also the localization of the range measurements, is considered to be solved by the Intel RealSense T265 tracking

camera, which estimates the robot's full six DOF pose based on visual simultaneous localization and mapping supported by an inbuilt inertial measurement unit[1]. The grid map is used as a model of the terrain geometry to identify passable places. It also captures the color of the terrain texture that is processed to compute the terrain appearance descriptors.

Algorithm 2. Spatial exploration.

We define the passability of the cell $v \in \mathcal{M}_{2.5D}$ as the probability $\pi(v)$ that the cell v can be traversed by the robot. The probability itself is based on the observed roughness of the terrain computed based on Bayer and Faigl (2021) as follows:

$$\rho(v) = \max_{v' \in 8nb(v)} \Delta(v, v'), \tag{7}$$

where $8\,nb(v)$ is the 8-neighborhood of the cell v, and the step height $\Delta(v_a, v_b)$ is as follows:

$$\Delta(v_a, v_b) = |\text{elevation}(v_a) - \text{elevation}(v_b)|, \tag{8}$$

where $\text{elevation}(v)$ denotes the estimated height of the terrain at v. The probability that the robot can pass a cell v is as follows:

$$\pi(v) \begin{cases} 0 & \text{if } \rho(v) > \rho_{\text{obstacle}} \\ 1 & \text{otherwise} \end{cases}, \tag{9}$$

where the threshold ρ_{obstacle} represents the lowest obstacle to be detected. An example of the grid map is shown in Figure 5A.

In active perception scenarios, the information about the terrain model $\mathcal{M}_{2.5D}$ gained by observing the cell v' is evaluated

---

1 In the simulated experiments, the localization is provided by the simulator.

57

FIGURE 5
Illustration of the color-geometric and cost models. **(A)** Visualization of the online built geometrical model with marked passability and clusters based on the cells with non-zero information according to the shown color legend; **(B)** terrain appearance descriptor calculated as a histogram of cell colors. The costs used in path planning; **(C)** minimal cost over gaits after the distance transform; **(D)** respective cheapest gait (gaits in red and purple). **(E)** Colors used to build the color histogram terrain appearance descriptor; **(F)** measured costs used for learning the GP (adjusted by hyperbolic tangent), visualized over the terrain appearance; **(G)** raw GP cost prediction; **(H)** GP prediction uncertainty. **(I)** Terrain clusters (arbitrary colors used to distinguish clusters); **(J)** information gained with terrain learning goals (goal colors corresponding to clusters); **(K)** cluster costs used in planning.

by entropy based on the known passability. Since the distribution of the passability is binary and depends on the 8-neighborhood of the cell, information gained by observing v' with unknown height is approximated as follows:

$$I_{\mathcal{P}}^{\text{cell}}(v') \approx \frac{k(v') + 1}{9}, \tag{10}$$

where $k(v)$ is the number of the unknown cells in the neighborhood of v. Thus, the expected information gained by perceiving the terrain from the position of the cell v can be expressed as follows:

$$I_{\mathcal{P}}^{\text{model}}(v) = \sum_{v' \in \delta(r_{\text{sensor}}, v)} \begin{cases} I_{\mathcal{P}}^{\text{cell}}(v') & \text{if observable}(v, v') \\ 0 & \text{otherwise} \end{cases}, \tag{11}$$

where $\delta(r_{\text{sensor}}, \nu)$ is the sensor range r_{sensor}-large neighborhood of ν; the function observable (ν, ν') returns `true` if and only if the cell ν' is observable from ν, which is determined by casting a ray from ν to ν' in the current elevation map $\mathcal{M}_{2.5D}$. Using all the cells with non-zero entropy in the TSP formulation is computationally intensive. Thus, we propose to spatially cluster the entropy to generate a limited number of spatial entropy representants by Algorithm 3.

```
Input: M_{2.5D} – Elevation grid map.
Output: Γ_P – Passability goal set.
1  Procedure cluster(M_{2.5D})
2      A ← ∅                                          // Init. set of clusters.
3      for ν ∈ M_{2.5D} : I_P^{model}(ν) > 0 do       // For each map cell with non-zero entropy.
4          if A = ∅ then                              // If no clusters in set.
5              A ← A ∪ {{ν}}                           // Create a new cluster.
6          else
7              d ← distanceToClosestCluster(ν, A)
8              if d < c_{radius} then
9                  addToClosestCluster(ν, A)          // Add point to existing cluster.
10             else
11                 A ← A ∪ {{ν}}                       // Create new cluster.
12     Γ_P ← ∅                                        // Init. cluster representants.
13     for A_i ∈ A do                                 // For each clusters.
14         if |A_i| > c_{min cells} then
15             Γ_P = Γ_P ∪ {averageCoordinateCell(A_i)}   // Create new representatives.
16     return Γ_P
```

Algorithm 3. Cluster entropy representatives

In addition to the terrain geometry, the grid map $\mathcal{M}_{2.5D}$ also carries the terrain texture calculated by the following approach. Each cell is provided a 10-bit color by projecting the camera image to the map $\mathcal{M}_{2.5D}$. Then the color space is shrunk to nine different colors, defined by color prototypes listed in Figure 5B. The relative amount of the cell colors within the radius r_{hist} matched to the selected color prototypes are used to build a 9-dimensional terrain appearance descriptor $\text{ta}(\nu)$ for each cell $\nu \in \mathcal{M}_{2.5D}$, which is visualized as a color histogram in Figure 5B.

4.3 Traversal cost model

The cost model $\mathcal{C}$ predicts the per-meter traversal cost c over observed areas deemed passable by the geometric passability model $\mathcal{P}$. The traversal cost model predicts the traversal cost from terrain appearance. Since the robot position is abstracted as the center of its circular footprint, the $\mathcal{C}$'s per-meter-cost predictions are conservative estimates that take into account all the cells on the footprint.

$$\hat{c}(\nu_a, \nu_b) = \max_{\nu' \in \delta(r_{\text{robot}}, \nu_a)} \hat{c}(\nu'), \qquad (12)$$

where $\delta(r, \nu)$ lists all cells within the r-radius of cell ν, and $\hat{c}(\nu)$ is the $\mathcal{C}$ cost estimate over cell ν. An example of the traversal cost assessment is depicted in Figure 5C.

The cost $\hat{c}$ is reported for the whole model set $\mathcal{C} = \mathcal{C}^{\mathbb{G}} = \{\mathcal{C}^g\}_{g \in \mathbb{G}}$, since it is the best gait-terrain cost.

$$\hat{c}(\nu) = \min_{g \in \mathbb{G}} \hat{c}_{\mathcal{C}^g}(\nu), \qquad (13)$$

where each gait-terrain cost $\hat{c}_{\mathcal{C}^g}$ is the prediction of the particular model $\mathcal{C}^g$. In addition, when navigating through the environment, the robot selects its gait w.r.t. the minimization in Eq. 13, as depicted in Algorithm 4. An example of gait selection is visualized in Figure 5D. A distance transform with c_{loss} per-meter-loss is used over the cell grid with the best-gait costs $\hat{c}(\nu)$ to dissuade the robot from navigating areas near terrain boundaries where frequent gait changes are likely.

```
Input: M_{2.5D} – Elevation grid map; ψ_E – Exploration path; ν^{robot}_{1,...,n} – Robot positions; g^{enforced} –
       Enforced sampling gait.
1  while exploration is running do
2      getLatest(M_{2.5D}, ψ_E, g^{enforced})
3      if g^{enforced} ≠ ∅ then                        // If the robot is forced to sample a gait-terrain model.
4          setGait(g^{enforced})                       // Use the particular gait.
5      else
6          g^{best} ← argmin_{g∈G} max_{ν'∈δ(r_{robot},ν_i^{robot})} ĉ_{C^g}(ν')   // Find the best gait for the robot position.
7          setGait(g^{best})                           // Use the particular gait.
8      walkAlong(ψ_E)                                  // Continue along the exploration path.
```

Algorithm 4. Navigate

Each gait-terrain model $\mathcal{C}^g$ comprises the cost regressor $\mathcal{R}$ and the terrain type clustering $\mathcal{T}$. In $\mathcal{R}$, we use GP regression to predict the traversal costs because it provides the predicted values and models the prediction uncertainty. Each traversal cost regressor $\mathcal{R}$ is learned from the learning set $\mathcal{L}$ of the paired terrain descriptors and the respective traversal costs observed when using the particular gait g that are depicted in Figure 5E and Figure 5F, respectively. The particular learned cost regressor $\mathcal{R}$ is used to predict the normal distribution of the traversal cost at queried terrain descriptor ta as follows:

$$\mathcal{N}\left(\hat{\mu}_c, \hat{\sigma}_c^2\right)(\text{ta}, \mathcal{R}) = \text{predict}(\text{ta}, \mathcal{R}). \qquad (14)$$

The cost prediction (visualized in Figure 5G) is the expected value.

$$\hat{c}(\text{ta}, \mathcal{R}) = E\left(\mathcal{N}\left(\hat{\mu}_c, \hat{\sigma}_c^2\right)(\text{ta}, \mathcal{R})\right) = \hat{\mu}_c(\text{ta}, \mathcal{R}), \qquad (15)$$

and the uncertainty of the prediction (shown in Figure 5H) is characterized by the differential entropy.

$$H\left(\mathcal{N}\left(\hat{\mu}_c, \hat{\sigma}_c^2\right)(\text{ta}, \mathcal{R})\right) = \frac{1}{2}\log\left(2\pi e \hat{\sigma}_c^2(\text{ta}, \mathcal{R})\right). \qquad (16)$$

The prediction uncertainty is used to approximate the information gain $I_{\mathcal{C}}$ associated with sampling the individual observed terrains, thus identifying areas the robot needs to visit to improve the traversal cost model.

The terrain type clustering $\mathcal{T}$ identifies the distinct terrain types (terrain descriptor clusters) in the environment. The terrain class set $\mathcal{T}$ is designed to be disjoint regarding the prediction model. Thus, sampling the traversal cost model at a cell corresponding to one terrain class provides no, or severely limited, information regarding the traversal cost model at a location corresponding to a different class. In particular,

following Pasolli and Melgani (2011), the classes are selected to be mutually distant in the terrain descriptor space. Each observed cell is assigned the closest terrain class as the closest class in the descriptor space.

$$T^*(\nu) = \text{argmin}_{T \in \mathcal{T}} \|\text{ta}(\nu), \text{ta}(T)\|, \tag{17}$$

where $\text{ta}(T)$ is the appearance assigned to the terrain class $T \in \mathcal{T}$. Since, on small terrain classes, it might not be possible to acquire enough samples to learn the traversal cost with sufficient certainty, we apply class erosion as described in Supplementary Appendix S1. The erosion output is the learning class assignment T and the planning class assignment $\hat{T}$. We avoid computing the cost prediction for every cell independently[2] and report the $\mathcal{C}^g$ prediction over a particular area as the cost to traverse over its respective terrain type.

$$\hat{c}_{\mathcal{C}^g}(\nu) = \begin{cases} \hat{c}\big(\text{ta}(\hat{T}(\nu)), \mathcal{R}\big) & \text{if } \hat{T}(\nu) \neq \varnothing, \\ c_{\max} & \text{otherwise,} \end{cases} \tag{18}$$

where the maximum cost $c_{\max}$ is reported for cells with no class (i.e., eroded) $\varnothing$.

The rest of this section describes how the traversal cost experience used to learn the models is measured, how the GP regressor is learned, and how the terrain type clustering is used to identify the locations where to improve the cost model.

4.3.1 Traversal cost measurement

The measured traversal cost describes the time needed to traverse between cells as $z^c(\nu, \nu')$. Since the distance between 2 cells is significantly lower than the robot stride length, the cost is smoothed over path segments (cell sequences) with a fixed duration. In particular, the per-meter cost c is continually measured as the inverted robot velocity v^{-1} over the path segment traversed by the robot in the last Δt s.

$$v^{-1}(\psi_s) = \frac{T(\psi_s)}{\|\psi_s\|}, \tag{19}$$

where $\|\psi_s\|$ is the length of the segment in meters and $T(\psi)$ is the measurement duration that is fixed to Δt. If the robot had not changed its gait on the segment, the cost is reported to the particular model $\mathcal{C}^g$ as the cost to traverse the midpoint of the segment as $z^c(\nu_{\lfloor|\psi_s|/2\rfloor}, \nu_{\lfloor|\psi_s|/2\rfloor+1})$. In addition, to remove potential cost spikes, the cost is further smoothed using a moving average window of the same (Δt) duration. Since the inverse velocity is unbounded and has both high values and high variance for a

stuck robot, the cost to be used by the predictor is transformed as follows:

$$c = c_{\text{high}} \tanh\left(\frac{1}{c_{\text{high}}} \frac{v^{-1}}{v_{\max}^{-1}}\right), \tag{20}$$

where the maximum robot velocity $v_{\max}$ (maximum from all $g \in \mathbb{G}$) scales the cost of the robot moving over an ideal terrain to 1, and the high cost c_{high}, which should only be experienced by a stuck robot, is used in the transform to bound the cost values.

4.3.2 Gaussian process traversal cost regressor

The employed GP regressor predicts both the prediction mean and variance making it suitable to model the prediction distribution as in (Eq. 14). Its description is dedicated to Supplementary Appendix S2 to make the study self-contained. GP regressor is learned only if there are at least $n_{\mathcal{L}}^{\min}$ learning pairs in $\mathcal{L}$ to avoid learning overconfident predictors at the beginning of the exploration. The learning is summarized in Algorithm 5.

Input: $\mathcal{M}_{2.5D}$ – Elevation grid map; $z^c_{1,\ldots,n}$ – Cost measurements.
Output: $\mathcal{L}$ – Learning set; $\mathcal{R}$ – Regressor; $\mathcal{M}_{2.5D}$ – Elevation grid map with measured cost assignments.

1 **while** exploration is running **do**
2   getLatest ($\mathcal{M}_{2.5D}$)
3    $\mathcal{M}_{2.5D} \leftarrow$ insertIfNovel ($\mathcal{M}_{2.5D}, z^c_i$)   // Save novel cost measurements to grid map.
4    $\mathcal{L} \leftarrow \varnothing$ // Initialize learning set.
5   **for** $\nu \in \mathcal{M}_{2.5D} : \exists c(\nu), \exists \text{ta}(\nu)$ **do**   // For each described grid map cell with measured cost.
6       $\mathcal{L} \leftarrow \mathcal{L} \cup (\text{ta}(\nu), c(\nu))$ // Add the cell to the learning set.
7   **if** $|\mathcal{L}| \geq n_{\mathcal{L}}^{\min}$ **then**   // If the learning set is large enough.
8       $\mathcal{R} \leftarrow$ learn($\mathcal{L}$)   // Learn the GP regressor.
9   reportLatest ($\mathcal{L}, \mathcal{R}, \mathcal{M}_{2.5D}$)

Algorithm 5. Traversal cost model learning.

The covariance function used in this work is the squared exponential kernel.

$$K(x, x') = \sigma_s^2 \exp\left(-\frac{1}{2} \frac{(x - x')^2}{l^2}\right), \tag{21}$$

where σ_s^2 is the output variance, and l is the length scale. We consider that the robot's cost and feature models have known ranges based on (Eq. 20) and the histogram descriptor, respectively. Therefore, similar to Karolj et al. (2020), the kernel hyperparameters l and σ_s^2 and GP's σ_ϵ^2 have fixed values that we consider to be dependent on the system parameters.

The GP is continually relearned when new observations using the particular gait g are experienced. The learning complexity can be bounded by $\mathcal{O}(n^4)$, where n is the number of training points. The size of the learning set $\mathcal{L}$ is limited by using at most one training point corresponding to each cell in $\mathcal{M}_{2.5D}$ and by storing measurements only when

---

2 In practice, for small environments, it is feasible to compute the prediction for every cell, and we do so for visualization as depicted in Figure 5G and Figure 5H.

they are novel and thus likely to improve the model. Hence, the relative traversal cost $c(\nu)$ experienced at cell ν is paired with the appearance descriptor $ta(\nu)$ of the respective traversed terrain, and when building the learning set $\mathcal{L}$, the model reports the pair $(ta(\nu), c(\nu))$ for each cell where both values are available.

Since the robot keeps only one measurement for each cell, each novel cost measurement $z^c(\nu, \nu')$ experienced when using the gait g is allocated to the grid map cell ν and its neighbors in $8 \, nb(\nu)$, and the traversal cost $c(\nu)$ at the cell ν is modeled using the Kalman filter with the estimated value and covariance as follows:

$$c_k = \frac{\sigma^2_{\text{sense}} c_{k-1} + \sigma^2_{k-1} z^c_k}{\sigma^2_{\text{sense}} + \sigma^2_{k-1}}, \quad \sigma^2_k = \frac{\sigma^2_{\text{sense}} \sigma^2_{k-1}}{\sigma^2_{\text{sense}} + \sigma^2_{k-1}}, \quad (22)$$

where z^c_k is the kth cost measurement at ν and σ^2_{sense} is its variance. The filter is initialized by the first cost observation z^c_0 at the respective cell, and the initial filter variance is σ^2_0.

In total, two cases are considered as situations when the cost is novel, and thus the model should be improved by storing the cost w.r.t. (Eq. 22): 1) when the prediction is erroneous and 2) when the prediction is uncertain. For the former, the cost experienced at the cell ν is accumulated if the measured cost z^c is out of the approximate 95% confidence interval $|\hat{\mu}_c(ta(\nu)) - z^c| > 2\hat{\sigma}_c(ta(\nu))$ of the prediction at ν. For the latter, the approximated information gain of the prediction is considered, and the robot accrues measurements when there is a potential of information gain $I_\mathcal{C}(T(\nu)) > 0$, which computation is described in the following paragraphs.

4.3.3 Terrain type clustering and goal identification

The traversal cost exploration goals $\Gamma_\mathcal{C}{}^g$ are selected by the robot as areas where the model can be improved and thus are the areas where the traversal cost model is uncertain. Each goal represents a terrain class where the robot can sample novel information about the cost model. The overall approach to goal identification is summarized in Algorithm 6.

TABLE 2 Gait parameterization.

Gait parameter/gait	Fast gait	Tall gait
Gait cycle duration (s)	1.10	2.90
Step height (m)	0.04	0.07
Maximum forward speed (ms^{-1})	0.05	$1.25e - 2$

Algorithm 6. Terrain type clustering, goal identification, and cost identification.

```
Input: M_{2.5D} – Elevation grid map; T – Terrain classes.
Output: M_{2.5D} – Elevation grid map with class assignments; T – Terrain classes.
1  Procedure cluster(M_{2.5D}, T)
2      A ← ∅                                              // Init. the adaptation dataset.
3      for ν ∈ M_{2.5D} : ∃ ta(ν) do                      // For each described cell on the grid map.
4          A ← A ∪ ta(ν)                                  // Add the descriptor to the adaptation set.
5      for ta ∈ draw(A, n^{IGNG}) do                      // For a randomly drawn subset of the adaptation set.
6          T ← adaptIGNG(T, ta)                           // Adapt the IGNG (Alg. 11).
7      for ν ∈ M_{2.5D} : ∃ ta(ν) do                      // For each described cell on the grid map.
8          T*(ν) ← argmin_{T∈T} ‖ ta(ν), ta(T) ‖         // Assign its terrain type.
9      M_{2.5D} ← erode(M_{2.5D})                          // Erode the classes over the grid map.
10     return M_{2.5D}, T
```

Algorithm 7. Cluster.

The clustering scheme presented in Algorithm 7 is based on the IGNG, described in Supplementary Algorithm S1, to make the study self-contained. In the neural gas, each neuron is a terrain prototype $ta(T)$ in the descriptor space that represents a terrain class T. When separating the classes, the intuition is that for exponential kernels, the length scale describes the range from the data where the model can reliably extrapolate, as used, for example, in Karolj et al. (2020). Hence, new classes are inserted into the neural gas when the distance from all prototypes exceeds $\sigma^{\text{IGNG}} = 2l$. The neural gas is constructed incrementally by repeated adaptation using the appearance descriptors in the environment, where the size of each adaptation batch is limited to n^{IGNG} descriptors that are randomly sampled from all the descriptors, and the yielded terrain classes can be seen in Figure 5I.

```
Input: R – Regressor; L – Learning set; M_{2.5D} – Elevation grid map; T – Terrain classes.
Output: T – Terrain classes with information gain assignments.
1  Procedure computeInformationGain(R, L, M_{2.5D}, T)
2      H_C^{GT} ← -∞                                       // Initialize the experienced-terrain uncertainty threshold.
3      for T' ∈ T : ∃ ν ∈ M_{2.5D}, T(ν) = T' do          // For each terrain class represented on the eroded grid.
4          if |T' ∩ L| > m_T then                          // If the class has enough ground truth measurements.
5              H_C^{GT} ← max(H_C^{GT}, H(σ̂_c²(ta(T'))))   // Adjust the experienced-terrain uncertainty threshold.
6      for T' ∈ T : ∃ ν ∈ M_{2.5D}, T(ν) = T' do          // For each terrain class represented on the eroded grid.
7          I_C(T') ← max(H(σ̂_c²(ta(T'))) - H_C^{GT}, 0)   // Compute the information gain.
8      return T                                            // Return the terrain classes with assigned information gains.
```

Algorithm 8. Compute information gain.

The terrain classes for which the cost model can be improved are identified using the cost regressor $\mathcal{R}$-predicted traversal cost distribution $\mathcal{N}(\hat{\mu}_c, \hat{\sigma}_c{}^2)(ta(T))$ at the class prototypes $ta(T)$. The traversal cost exploration goals are selected according to Algorithm 8 as the classes where there is potential for

```
Input: M_{2.5D} – Elevation grid map; R – Regressor; L – Learning set.
Output: M_{2.5D} – Elevation grid map with cost assignments Γ_C – Cost goals; T – Terrain classes.
1  T ← ∅                                                  // Init. terrain class set.
2  Γ_C ← ∅                                                // Init. goal set.
3  while exploration is running do
4      getLatest(M_{2.5D}, R, L)
5      if R ≠ ∅ then
6          M_{2.5D}, T ← cluster(M_{2.5D}, T)              // Update terrain clusters (Alg. 7).
7          T ← computeInformationGain(R, L, M_{2.5D}, T)  // Compute information gain (Alg. 8).
8          Γ_C ← identifyGoals(M_{2.5D}, T)               // Identify goals (Alg. 9).
9          M_{2.5D} ← setPlanningCost(M_{2.5D}, T)         // Identify costs (Alg. 10).
10         reportLatest(M_{2.5D}, Γ_C, T)                 // Report costs assigned to M_{2.5D}, goals, and class set.
```

TABLE 3 System parameterization.

Symbol	Parameter	Unit	Value, split by environment	
			Real/Small Sim.	Large Sim.
d_v	Grid map cell size	m	0.05	0.10
r_{sensor}	Sensor range	m	2.5	10.00
c_{radius}	Spatial clustering radius	m	0.50	2.00
c_{min_cells}	Spatial clustering, min cells per cluster	-	10	10
r_{robot}	Robot footprint radius	m	0.25	0.40
$\rho_{obstacle}$	Roughness passability threshold	m	0.25	0.25
r_{hist}	Histogram descriptor radius	m	0.25	0.30
Δt	Cost measurement window duration	s	5.00	1.00
v_{max}	Maximum robot velocity	m s^{-1}	0.05	0.25
c_{loss}	Cost distance-transform per-meter loss	–	10.00/15.00*	7.5
c_{high}	High cost for cost transform	–	20.00	20.00
c_{max}	Maximum cost for path planning	–	20.00	20.00
σ_{sense}	Kalman filter cost measurement variance	–	0.10	0.10
σ_0^2	Kalman filter initial variance	–	1.00	1.00
σ_s	GP output variance	–	1.00	1.00
σ_ϵ	GP observation noise	–	0.50	0.50
l	GP length scale	–	0.40	0.40
$n_\mathcal{L}^{min}$	Minimum learning set size	–	25.00	25.00
n_{erode}^{steps}	Cluster erosion steps	–	2.00	2.00
m_T	Minimum size of a ground truth cluster	–	10.00	10.00
d_S	Cost-model sampling lattice cell size	m	0.44	0.44
n_{dilate}^{steps}	Cluster dilation steps	–	3.00	3.00
n_{dilate}^{size}	Cluster dilation size	–	2.00	2.00
ϵ_1^{IGNG}	GNG warp scale winner	–	$1.00e-3$	$1.00e-3$
ϵ_{nb}^{IGNG}	GNG warp scale neighbor	–	$1.00e-5$	$1.00e-5$
a_{mature}^{IGNG}	GNG age mature	–	$1.00e2$	$1.00e2$
a_{max}^{IGNG}	GNG max edge age	–	50.00	50.00
n^{IGNG}	GNG learning batch size	–	$5.00e3$	$5.00e3$
Δt_{sample}	Cost sampling duration	s	30.00	12.00
$\Delta t_{fallback}$	Stuck fallback duration	s	30.00	3.00

* Different value used in *small* simulation/real deployment.

TABLE 4 System operation frequencies.

| Module	Frequency	Condition
Elevation mapping	5.00 Hz	
Spatial goal identification	0.33 Hz	
Cost measurement	20.00 Hz	Only if using the respective gait
Cost learning	0.10 Hz	Only if not already running
Goal identification	0.10 Hz	
Goal sequence planning	1.00 Hz	Only after goal set change or reaching a goal
Path planning	1.00 Hz	Only after goal set change or reaching a goal
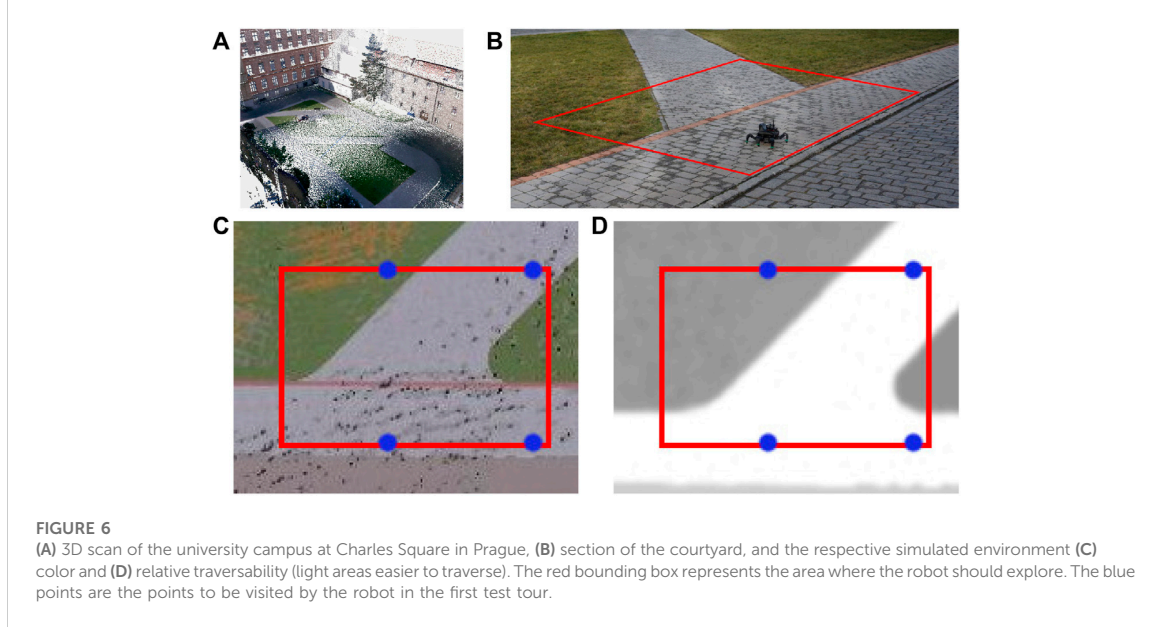
FIGURE 6
(A) 3D scan of the university campus at Charles Square in Prague, **(B)** section of the courtyard, and the respective simulated environment **(C)** color and **(D)** relative traversability (light areas easier to traverse). The red bounding box represents the area where the robot should explore. The blue points are the points to be visited by the robot in the first test tour.

information gain; see the visualization in Figure 5J. The gain is approximated from the prediction entropy.

$$I_\mathcal{C}(T) \approx \max\big(H\big(\hat{\sigma}_c{}^2(\mathrm{ta}(T))\big) - H_\mathcal{C}^{\mathrm{GT}}(\mathcal{L}), 0\big), \qquad (23)$$

where $H_\mathcal{C}^{\mathrm{GT}}$ is a threshold value associated with the uncertainty of the experienced traversal costs. The robot learns when there is potential of information gain $I_\mathcal{C} > 0$, and no information can be gained at eroded cells $I_\mathcal{C}(\varnothing) = 0$. We set the threshold value based on the highest prediction uncertainty for terrains that are considered certain since they cover cells that are already in the learning set as follows:

$$H_\mathcal{C}^{\mathrm{GT}}(\mathcal{L}) = \max_{T \in \mathcal{T}:\, |\{v \in \mathcal{M}_{2.5D}:\, T(v)=T\} \cap \mathcal{L}| > m_T} H\big(\hat{\sigma}_c{}^2(\mathrm{ta}(T))\big), \qquad (24)$$

where we avoid overconfident GP-predictions for barely sampled terrains by allowing only terrain classes with at least m_T observed ground truth cost values. The threshold equals the maximum value over such ground truth terrain classes.

The sampling locations (visualized, for example, in Figure 5J) corresponding to the terrain class are sampled along a lattice S with the cell size $d_S \gg d_v$, as depicted in Algorithm 9. For each lattice point p_S, the closest cell v in $\delta(\frac{\sqrt{2}\,d_S}{2}, p_S)$ radius that is not associated with a traverability measurement and that is informative with $I_\mathcal{C}(T(v)) > 0$ is reported as a sampling site; if no such cell exists, no site is reported for the lattice point. Since only cells without measurements are considered, it is possible for small terrain classes to run out of cells before reaching m_T measurements. In such a case, the class is considered too small to learn and is no longer reported as a goal, and it is pruned from the class set. In addition to the goals, the traversal cost $\hat{c}_{\mathcal{C}^g}(v)$ (visualized in Figure 5K) is also reported for the v's prototype $\mathrm{ta}(\hat{T}(v))$ w.r.t. (Eq. 13) according to Algorithm 10.

Input: $\mathcal{M}_{2.5D}$ – Elevation grid map; $\mathcal{T}$ – Terrain classes.
Output: $\mathcal{M}_{2.5D}$ – Elevation grid map with cost assignments.
1 **Procedure** setPlanningCost $(\mathcal{M}_{2.5D}, \mathcal{T})$
2    $\mathcal{M}_{2.5D} \leftarrow \mathrm{dilate}(\mathcal{M}_{2.5D})$ // Dilate the classes over the grid map.
3   **for** $v \in \mathcal{M}_{2.5D}$ **do**   // For each cell.
4   **if** $\hat{T}(v) \neq \varnothing$ **then**   // If the cell has a dilated class.
5        $\hat{c}_{\mathcal{C}^g}(v) \leftarrow \hat{c}(\hat{T}(v))$ // Report the class cost.
6   **else**   // Otherwise.
7        $\hat{c}_{\mathcal{C}^g}(v) \leftarrow c_{\max}$ // Report the maximum cost.
8   **return** $\mathcal{M}_{2.5D}$ // Return the map with planning cost assignment.

Algorithm 10. Set planning cost.

5 Experimental evaluation

The proposed exploration with active terrain learning has been examined in simulated trials and real experimental

---

Input: $\mathcal{M}_{2.5D}$ – Elevation grid map; $\mathcal{T}$ – Terrain classes.
Output: $\Gamma_\mathcal{C}$ – Cost model goals.
1 **Procedure** identifyGoals $(\mathcal{M}_{2.5D}, \mathcal{T})$
2   **for** $T \in \mathcal{T}: I_\mathcal{C}(T) > 0$ **do**   // For each terrain class where information can be gained.
3      $\gamma_\mathcal{C}(T) \leftarrow \varnothing$ // Initialize the sampling site set.
4   **for** $v \in S$ **do**   // For each cell on the sampling lattice.
5   **if** $\exists v' \in \mathcal{M}_{2.5D}: \|v, v'\| < \frac{\sqrt{2}}{2}d_S, I_\mathcal{C}(T(v')) > 0, c(v') = \varnothing$ **then**   // If there is a close enough cell that has non-zero inforation gain and no measured cost.
6        $v'' \leftarrow \mathrm{argmin}_{v' \in \mathcal{M}_{2.5D}: \|v,v'\| < \frac{\sqrt{2}}{2}d_S, I_\mathcal{C}(T(v')) > 0, c(v') = \varnothing} \|v, v'\|$ // Find the closest such cell.
7        $\gamma_\mathcal{C}(T(v'')) \leftarrow \gamma_\mathcal{C}(T(v'')) \cup v''$ // And add it to the respective goal as a sampling site.
8   **for** $T \in \mathcal{T}: I_\mathcal{C}(T) > 0, |\gamma_\mathcal{C}(T)| = 0$ **do**   // For each terrain class with information gain but no sampling cell.
9      $\mathcal{T} \leftarrow \mathcal{T}/T$ // Prune the terrain class.
10   **return** $\cup_{T \in \mathcal{T}: I_\mathcal{C}(T) > 0} \gamma_\mathcal{C}(T)$ // Return the goal set.

Algorithm 9. Identify goals.

FIGURE 7
Environment assessment after the simulated scenario run with regards to both gaits; **(A)** dominant color in the histogram feature; **(B)** merged cost used for planning; **(C)** selected gait (fast in red, tall in purple); **(D)** costs used for learning the fast gait model [adjusted by hyperbolic tangent in (Eq. 20)], visualized over the terrain appearance; **(E)** clusters used in the fast gait model (arbitrary colors used to distinguish clusters); **(F)** fast gait cost predictions assigned by the dilated clusters; **(G)** costs used for learning the tall gait model [adjusted by hyperbolic tangent using (Eq. 20)], visualized over the terrain appearance; **(H)** clusters used in the tall gait model (arbitrary colors used to distinguish clusters); **(I)** tall gait cost predictions assigned by the dilated clusters; **(J)** exploration run; **(K)** test-tour run using the baseline model without the learned traversal costs; **(L)** test-tour run using the learned traversal costs. The development of the path through the fully discovered simulated environment during the exploration; **(M)** at the beginning of the exploration, the robot uses flat costs and thus does not avoid difficult terrains; **(N)** after learning the costs for the fast gait, the robot is too cautious and avoids going near the costly turf; **(O)** after learning the tall gait costs, the robot is less cautious and is willing to walk near difficult terrain.

deployments using a hexapod walking robot. The simulated and real scenarios have been set up so that the robot first explores the environment and learns the cost models using the proposed method and, in some tests, a selected baseline method. Then the performance has been evaluated and compared with the baseline approach by navigating the robot over a sequence of benchmark waypoints using the respective traversal cost models of the environment learned during the exploration.

The hexapod walking robot, which can be seen in Figure 1, is used in the real deployment, and the simulation is parameterized to mimic the robot's motion and sensory capabilities. The robot has six legs, each comprising three Dynamixel XM430-W350 servomotors. The robot is equipped with the Intel RealSense D435 camera used to construct the colored environment model and the Intel RealSense T265 localization camera. The onboard computation is provided by the Intel NUC 10i7FNK with Intel Core i7 10710U accompanied with 64 GB memory, running Ubuntu 18.04 with ROS Melodic (Quigley et al., 2009). The robot locomotion is facilitated by a blind adaptive motion gait (Faigl and Čížek, 2019). The robot uses two particular gait configurations, see Table 2: The fast gait suitable for flat, even

TABLE 5 Performance as the time (total cost) in seconds to traverse.

Small virtual environment, tour 1 (mean ± std of 25 runs)

Method/Time [s]	Segment 1	Segment 2	Segment 3	Segment 4	Full tour
Baseline	79.99 ± 0.00	239.59 ± 6.62	133.20 ± 6.76	177.59 ± 13.04	630.39 ± 21.06
Gait selection	80.00 ± 0.00	275.00 ± 8.06	125.49 ± 7.39	164.00 ± 7.39	644.50 ± 7.34
Proposed	80.00 ± 0.00	119.99 ± 0.00	112.40 ± 4.27	142.40 ± 4.27	454.80 ± 4.27

Small virtual environment, tour 2 (25 runs)		Small virtual Environment, exploration (5 runs)	
Method/Time [s]	Full tour	Environment	Time [s]
Baseline	2748.00 ± 30.59	GTSP	1382.68 ± 241.47
Gait selection	2523.12 ± 39.48	Greedy	1547.16 ± 203.71
Proposed	2271.99 ± 33.38		

Large virtual environment (mean ± std of 5 runs)			Real deployment	
Method	Full tour time [s]	Exploration time [s]	Test	Time [s]
Proposed	554.00 ± 13.56	1167.15 ± 163.69	Test Segment, baseline	454.00
Spatial-only	859.99 ± 156.02	545.40 ± 137.43	Test Segment, proposed	143.00
			Exploration, proposed*	1364.00

* The similarity between the real and simulated times to explore is coincidental.

surfaces, and the tall gait that performs better than the fast gait over rough terrain but otherwise is slower. The robot is equipped with a reflex that detects that the robot is stuck with costs exceeding $c_{\max}$ and switches over to the tall for $\Delta t_{\text{fallback}}$ seconds to avoid the robot getting stuck when using the baseline model or at the beginning of the learning process. The parameterization of the proposed method can be found in Table 3, and the operating frequencies of the proposed method's processes are depicted in Table 4.

5.1 Simulated scenarios

The simulated scenarios are based on a courtyard environment captured by four 3D scans obtained using Leica BLK 360 3D scanner and visualized in Figure 6A. The scanner has standard deviation of 4 mm at 10 m and 7 mm at 20 m. The scans total approx. 1.4×10^8 points.

In total, two virtual environments are created using the scan: small and large. The small environment represents a small section of the courtyard, where the simulated robot mimics the real robot's speed and sensory equipment. It is used to test the benefit of the individual components of the proposed approach by comparing them to baseline methods where the particular component is removed or simplified. The large environment comprises terrain segments observed in the scan that are rearranged to create a larger, artificial environment with obstacles where different exploration algorithms are compared using a faster robot with an extended sensor range.

5.1.1 Small environment

The small environment is concerned with a section of the environment that is detailed in Figure 6B. We have created a simulation model of the environment containing several types of pavement (gray and red) and turf (green, brown), shown in Figure 6C. The turf is modeled as hard to traverse and can get the robot stuck for the fast gait, whereas the pavement does not impede the robot, see Figure 6D.

First, to demonstrate the benefits of using a cost model learned from prior experience, the robot is tasked to execute two tours in the environment using the learned cost model and a flat-cost baseline model. Second, the utility of exploring along the proposed GTSP-derived path is demonstrated by comparing its time to explore the environment with a greedy, myopic baseline, which drives the robot to the cheapest goal to reach w.r.t. the so far learned costs.

The first tour comprises four waypoints. The robot starts at the bottom-left point and executes the tour counter-clockwise until reaching the start location again. The two particular areas are designed to demonstrate the utility of the learned model: 1) the segment between the bottom-right and top-right waypoints where the robot can choose either a direct route over the turf or a longer path over the pavement and 2) the area around the top-left waypoint where the turf cannot be avoided and thus the robot needs to switch to the tall gait. The second tour comprises 20 points randomly sampled in the environment, and it serves to demonstrate the performance of the learned model over a tour that was not handcrafted.

FIGURE 8
Large simulated environment **(A)** color and **(B)** relative traversability, **(C)** and the test tour through the environment, which starts at the starred node and is counter-clockwise. The built maps of the large simulated environment: **(D)** geometric map and **(E)** merged costs used for planning after exploration using the proposed approach; merged costs ofter exploration using the spatial-only model while **(F)** avoiding and **(G)** traversing rough terrain, respectively.

In addition to the proposed approach and the baseline, in the simulated tests, we also deploy a hybrid gait selection approach that chooses its gait using the proposed model but does not plan its path w.r.t. the predicted costs and walks directly to the next waypoint. Unlike the baseline approach, which switches to the tall gait when stuck and repeatedly tries to switch back to the fast gait, the hybrid gait selection approach switches gaits only when approaching or leaving the terrain identified as hard to traverse by the model. Hence, it should outperform the baseline over longer sections on difficult terrains, where the baseline is slowed down by trying to switch back to the fast gait.

The simulation environment consists of the Intel i7-9700 3.00 GHz with 32 GB memory running Ubuntu 18.04 with ROS

Melodic. Since the captured environment comprises terrains that might slow down the robot because they are somewhat non-rigid, instead of using a geometry-based simulator such as Gazebo, which cannot model such terrains, we elect to build a virtual environment over a simple simulator using real-world data. The simulation is performed using the simple two dimensional robot simulator (STDR)[3] within the ROS ecosystem. On top of the simulator, we have implemented an interface that simulates the robot's RGB-D camera, which assigns each point in the robot's simulated exteroceptive measurements color based on the point's

---

3 http://stdr-simulator-ros-pkg.github.io

position in the environment color map shown in Figure 6C and filters the measurements to contain only points within the 87 deg wide field of view of the simulated RGB-D camera. The terra-mechanical properties are simulated by slowing down the robot over the individual traversed terrains w.r.t. the performance observed over such terrain in a real-world deployment, as shown in Figure 6D.

In the evaluation, the robot first explores and learns the models shown in Figure 7A to Figure 7I. An example exploration path can be seen in Figure 7J. The robot learns that the turf, which appears either green or brown, cannot be traversed by the *fast* gait and thus selects the *tall* gait over that terrain type. On the other hand, the pavement does not hinder the *fast* gait, which is considerably faster and thus preferred.

Although the two gait models create the terrain clusters independently, the clusters in Figure 7E and Figure 7H differ only in cluster indices used in the internal representation (each index is associated with a different color in the visualization). It can be observed that the robot does not use any clusters associated with the red line on the pavement, either removing the thin cluster outright in the erosion or pruning the small erosion remains after the robot finds out that it cannot get enough samples to learn such a small terrain.

In the particular exploration run shown in Figure 7J, the robot first walks along the left side of the exploration bounds, learning the fast gait costs for both the pavement and turf and the tall gait cost over the turf. Then the robot learns the tall gait cost over the pavement while clearing the spatial exploration goals. During the exploration, it can be seen that the robot avoids walking over the remaining turf, only approaching it at the very end of the exploration. Thus, the robot needs only to enter and not leave the turf (minimizing the time on the costly terrain) to reach the goal that lies on the turf.

The test runs using the baseline, and the learned model over the first tour are shown in Figure 7K and Figure 7L, respectively. In addition, the development of the tours that would be used at different points during the exploration can be seen in Figure 7M through Figure 7O. In the baseline test, the robot walks directly between the waypoints and only switches to the *tall* gait after getting stuck. On the other hand, when using the learned model, the robot avoids the turf if possible and switches to the tall gait before entering the turf while pursuing the top-left goal.

The performance over 25 simulated trials (five exploration runs, each with five tour tests for the tour tests; 25 runs for the simulated exploration tests) can be observed in Table 5. On the first tour, the hybrid gait selection approach is slower than the reactive baseline. In the authors' opinion, it is caused by the conservative (large) value of r_{robot}, which compels the robot to use the slow tall gait on the border between the rough terrain and pavement, whereas the reactive approach only tries to switch back to the fast gait (which is its main disadvantage when compared to the hybrid approach) a few times on the short rough terrain segment. Nonetheless, the proposed learned model knows to avoid such areas and performs better or the
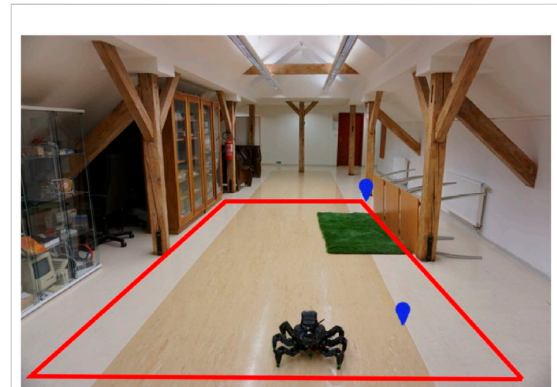
FIGURE 9
The 2m × 6m large deployment area with a green artificial turf. The area boundary is in red, and the waypoints of the test tour are depicted in blue. The shown robot is at the starting position.

same as the other approaches over every tour segment. Hence, the results suggest that robot benefits from using the learned costs in path planning. Over the second tour, the robot performs similarly. The learned model outperforms the baseline when moving around or over the turf. Both approaches exhibit similar travel times when the direct path between the waypoint leads only over the pavement. Unlike over the first tour, the hybrid gait selection performs better than the baseline approach, presumably due to longer sections over hard-to-traverse terrains on the second tour. The proposed approach consistently outperforms the baseline and hybrid gait selection approaches; we conclude that the robot benefits from using the learned model.

In addition to the tour tests, the results suggest that the robot benefits from using the non-myopic GTSP planner compared to the myopic greedy approach. Even though the performance of the two approaches appears relatively close, the Mann–Whitney U Test (Mann and Whitney, 1947) rejects the null hypothesis of the same exploration time distribution at 99.5% confidence against both the two-sided and the relevant one-sided alternative. In the authors' opinion, the high variance in the observed exploration times can be attributed to the effect of random chance in exploration since neither myopic nor non-myopic approaches are informed about the terrains in unexplored areas. However, the myopic explorer is more likely to make a bad decision, such as not clearing some of the goals in a particular area that needs to be visited later. Therefore, the proposed non-myopic approach performs better overall.

5.1.2 Large environment

The large environment is an artificial 20 × 25 m outdoor/indoor scenario. The map comprises patches from the courtyard
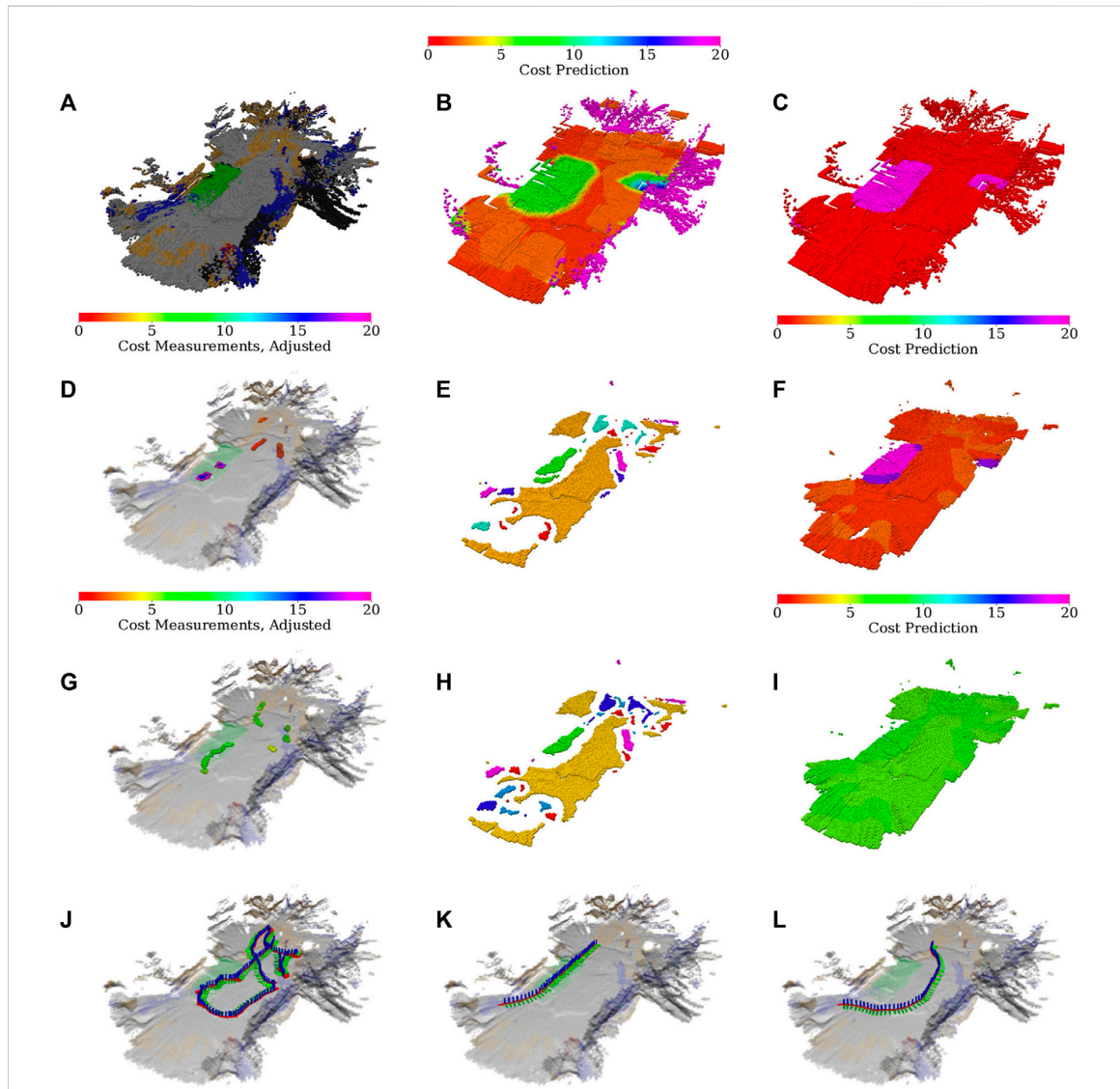
FIGURE 10
Environment evaluation and the real robot exploration run; **(A)** dominant color in the histogram feature; **(B)** merged cost used for planning; **(C)** selected gait (*fast* in red, *tall* in purple); **(D)** costs used for learning the *fast* gait model (adjusted by hyperbolic tangents), visualized over the terrain appearance; **(E)** clusters used in the fast gait model (arbitrary colors used to distinguish clusters); **(F)** fast gait cost predictions assigned by the dilated clusters; **(G)** costs used for learning the tall gait model (adjusted by hyperbolic tangents), visualized over the terrain appearance; **(H)** clusters used in the tall gait model (arbitrary colors used to distinguish clusters); **(I)** tall gait cost predictions assigned by the dilated clusters; **(J)** exploration run; **(K)** test-tour run using the baseline model without the learned traversal costs; **(L)** test-tour run using the learned traversal costs.

scan rearranged as shown in Figure 8. Given the size of the environment, the robot is sped up five times. The cell size is increased to 0.1 m, and other parameters are adjusted accordingly, see Table 3. In addition, the robot uses an omnidirectional sensor with the increased range of 10 m, which expands the range of terrains that can be observed without the respective terrain's traversal. To accommodate the

simulation of the increased sensor range, the virtual environment is run on AMD Ryzen Threadripper 3960× 3.8 GHz with 48 GB memory running Ubuntu 18.04 and ROS Melodic, using STDR in the same manner as for the small environment.

Similar to the small environment, the robot is first set to explore the environment and then is tasked to visit the set of waypoints shown in Figure 8C. The proposed algorithm is

compared to a spatial-only baseline approach, which learns the cost models only as a result of experiencing cost while pursuing spatial exploration goals. The spatial-only changes the gaits in a reactive fashion when stuck and hence only learns the model for the tall gait if it enters the difficult green or brown turf during the exploration.

The quantitative results for the large environment are shown in Table 5. Since the proposed approach actively tries to sample every terrain type, it is slower to explore the whole environment. However, the proposed approach performs better in the tour evaluation. Closer examination suggests that while the tour times of the proposed approach remain similar in all trials, the spatial-only times vary wildly since the learned models differ based on which terrains the robot has traversed during the exploration. This randomness can be attributed to differences in simulation and plan execution. In addition, Figures 8D–G shows the learned maps for the proposed model, and for the spatial-only model in both the cases when the rough terrain was and was not traversed. For the case when a rough terrain was traversed by the spatial-only model, the costs differ between the individual rough areas. However, the ground truth costs shown in Figure 8B suggest that they should be the same, as is the case for the proposed model. Likely, this is caused by the robot traversing only the brown-green rough terrain located on the left of the environment. The green terrain, located in the center and right of the environment, appears somewhat similar to the brown-green terrain. Hence, the robot considers it to be difficult to traverse to a certain degree. However, since the spatial-only model does not deliberately sample the terrains, the model's guess differs somewhat from the exact cost to traverse the particular terrain, decreasing the fidelity of the predictions.

Overall, the presented results suggest that the proposed approach presents a tradeoff in terms of exploration and execution time: the longer time spent exploring the environment and learning the cost models provides the robot with better cost maps, which shorten the time to navigate the environment after it is explored. It should be noted that since the behavior of the spatial-only model is affected by random chance (differences in simulation and plan execution), it can provide models as good as the proposed approach. However, there is no guarantee that this would happen regularly, whereas the proposed approach has returned high fidelity maps in every test case.

5.2 Real robot experimental deployment

The viability of the proposed approach is demonstrated in the real experimental deployment, where the robot explores an indoor 2 × 6 m area visualized in Figure 9. The office-like environment comprises flat synthetic terrain that is easy to

traverse but appears to the robot differently colored at different locations since it is glossy and carries the color of nearby objects located next to the arena. When building the colored elevation map $\mathcal{M}_{2.5D}$, we use the first color observed at each location to account for the issue. In addition to the flat terrain, a green artificial turf is placed in a part of the test area to provide a relatively hard terrain to traverse. The robot interacts with the real terrains similarly to the simulation: the *fast* gait may get stuck on the turf but is faster than the *tall* gait over the flat parts of the arena. During the experiment, the robot is set to explore the area; even though it can leave the bounds of the 2 m × 6 m large area, it does not pursue goals located outside of the bounds.

Figure 10 shows the maps learned in the experimental run, which is also presented in the accompanying Supplementary Video S1. A colored map of the environment is depicted in Figure 10A. The overall costs and selected gaits through the environment are shown in Figure 10B and Figure 10C, respectively.

During the experimental deployment, the robot first learns the largest gray appearing flat terrain using the fast gait. Then it learns on the turf for both gaits and returns to the gray area to learn for the tall gait. Afterward, the robot pursues the yet unvisited spatial goals and smaller off-color terrain clusters that appear near the environment boundary and are caused by the glossy floor that carries the color of the nearby objects.

Compared to the simulation, the robot needs a larger amount of the measurements to learn the terrains (see Figure 10D and Figure 10G), and there are more terrain clusters (see Figure 10E and Figure 10H). It suggests that the real environment is noisier and contains multiple differently colored areas, which is in line with our observations regarding the glossy floor material. Nevertheless, the traversal costs learned by the robot for the individual gaits (see Figure 10F and Figure 10I) are within expectations, as is the overall planning cost depicted in Figure 10B and gait selection visualized in Figure 10C.

The test run scenarios are set up similarly to the tours used in the simulated test; the robot is placed in front of the hard-to-traverse turf and tasked to reach a goal location behind the hard-to-traverse terrain, slightly out of the exploration bounds, see Figure 9. The paths shown in Figure 10K and Figure 10L show that when using the baseline without the learned model, the robot tries to reach the goal directly over the turf, gets stuck, and needs to switch to the slow *tall* gait. On the other hand, when using the learned model, the robot avoids the hard-to-traverse areas and reaches its goal quickly using the fast gait. The performance in the presented run can be seen in Table 5. Overall, we conclude that the real deployment confirms that the robot can actively learn the traversability as a part of the exploration mission and benefits from using such learned models.

6 Discussion

The presented exploration system is proposed as a combination of spatial geometric modeling and learning terrain-gait traversal cost models. However, the system is designed to support additional models that do not describe the robot's traversal cost. Moreover, since the models are kept separate, there is no need to use the same feature set for each of them. Therefore, the approach is compatible with spatial models such as magnetism models (Karolj et al., 2020) or GP-based occupancy (Wang and Englot, 2016). The only requirement for a model is that it produces a set of learning goals in the environment that are resolved once particular information is sampled. Hence, the proposed system can be extended by including additional traversability models, such as modeling the passability of potentially non-rigid obstacles.

In addition, we approach the traversal cost prediction so that it supports any cost model that is additive along the traversed path, such as time to traverse or consumed energy. Besides, individual cost predictors describe the gaits of a hexapod walking robot, but they can also describe any discrete set of robot configurations. Hence, the approach is viable for any mobile robot that describes its motion experience using an additive cost and can also be used to model the energy a tracked robot consumes, for example, with adjustable flippers. A particular limitation of the cost modeling used in the presented approach is that we assume that the individual gaits are switched for free w.r.t. the cost (i.e., instantaneously for cost modeled as the time to traverse), whereas in practice, the gait requires some time to exhibit its properties. In this study, we leave the question of how to predict gait-change cost open for future work.

The used cost model goal generation stems from the idea that adding new observations does not increase GP uncertainty if the hyper-parameters are fixed (Rasmussen and Williams, 2006). Therefore, sampling new measurements should not increase uncertainty and thus not spawn new goals in areas containing none. In practice, even though we use fixed GP hyper-parameters, the non-increasing nature of the uncertainty does not strictly hold for the approximated information gain since, in addition to the GP hyper-parameters, the information gain also depends on the terrain clusters and the costs and descriptors in the learning set, all of which might drift during the exploration. However, the robot behavior demonstrated in both evaluation setups shown in Figure 7J and Figure 10J suggests that the assumption holds in general. The robot clears the areas corresponding to the individual terrains (goals) and is not compelled to return to previously visited areas.

The primary limitation of the proposed approach is identified in its inability to compare the utility of the goals originating from the different models. We are motivated to build a modular system that would support different model types; therefore, the proposed decoupled approach considers each goal equally valued, regardless of the source model. This limits how the models are used since the goal utility, such as the information gain, is relegated to be used only inside the particular model to determine which environment features (locations or terrain types) are goals to use in creating an instance of the GTSP. The proposed approach provides a non-myopic solution to visit the goals reported by the individual models, where the models are also non-myopic since each can report multiple goals. Myopic models that would report their respective highest utility goal (potentially with multiple sampling sites) can be used. However, similarly to the myopic planner with the results reported in Table 5, the time to explore would likely increase since the GTSP planner would lack the information on where to go after the current goals are sampled, and thus the exploration path would often change significantly. Integrating goal utility into the decoupled planning and using alternative utility functions such as the GP-UCB remains the subject of future work.

7 Conclusion

In this study, we present a system for autonomous mobile robot exploration that incorporates active learning of traversal cost models in addition to spatial model building. During the exploration, the robot builds the spatial geometric model of the environment and learns the traversal cost models, each comprising a Gaussian process regressor and a growing neural gas terrain clustering scheme. The geometric model is used to determine areas passable by the robot, while the cost models predict the traversal costs over the passable terrains from the terrain's appearance. Each cost model corresponds to a particular hexapod walking robot locomotion gait. The robot approaches exploration in a decoupled manner, creating a set of goals for the spatial exploration and for each traversal cost model. The exploration path is planned by solving an instance of the generalized traveling salesman problem over the goals that are sets of possible sites of visits to improve the particular model. The proposed system has been evaluated in simulation setup and real experimental deployment with two different walking gaits. The results suggest that the proposed system yields the robot to explore the environment and learn the traversal cost models. The learned models benefit the robot's operation in the environment. In future work, we plan to model the gait change costs, include additional traversability models such as obstacle rigidity, and extend the proposed approach to support goal utility and exploration–exploitation models.

Data availability statement

The raw data supporting the conclusion of this article will be made available by the authors, without undue reservation.

70

Author contributions

With the support of JF, MP, and JB designed the proposed system. MP and JB performed the experiments and processed the data. MP, JB, and JF wrote the manuscript. All the authors contributed to the manuscript and approved the submitted version.

Funding

Acknowledgments

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/frobt.2022.910113/full#supplementary-material

References

Azpúrna, H., Campos, M. F. M., and Macharet, D. G. (2021). Three-dimensional terrain aware autonomous exploration for subterranean and confined spaces. *IEEE Int. Conf. Robotics Automation (ICRA)*, 2443. –2449. doi:10.1109/ICRA48506.2021.9561099

Baleia, J., Santana, P., and Barata, J. (2015). On exploiting haptic cues for self-supervised learning of depth-based robot navigation affordances. *J. Intell. Robot. Syst.* 80, 455–474. doi:10.1007/s10846-015-0184-4

Bayer, J., and Faigl, J. (2021). "Decentralized topological mapping for multi-robot autonomous exploration under low-bandwidth communication," in *European Conference on Mobile Robots* (Bonn, Germany: ECMR), 1–7. doi:10.1109/ECMR50962.2021.9568824

Bayer, J., and Faigl, J. (2019). "Speeded up elevation map for exploration of large-scale subterranean environments," *In 2019 Modelling and Simulation for Autonomous Systems* (Palermo, Italy: MESAS), 192–202. doi:10.1007/978-3-030-43890-615

Bayer, J., and Faigl, J. (2020). "Speeded up elevation map for exploration of large-scale subterranean environments," in 2020 Modelling and Simulation for autonomous systems (MESAS). *190–202*.

Bekhti, M. A., and Kobayashi, Y. (2016). "Prediction of vibrations as a measure of terrain traversability in outdoor structured and natural environments," in *Image and video technology*, 282–294. doi:10.1007/978-3-319-29451-3_23

Belter, D., Wietrzykowski, J., and Skrzypczyński, P. (2019). Employing natural terrain semantics in motion planning for a multi-legged robot. *J. Intell. Robot. Syst.* 93, 723–743. doi:10.1007/s10846-018-0865-x

Binney, J., and Sukhatme, G. S. (2012). Branch and bound for informative path planning. *IEEE Int. Conf. Robotics Automation (ICRA)*, 2147. –2154. doi:10.1109/ICRA.2012.6224902

Bourgault, F., Makarenko, A. A., Williams, S. B., Grocholsky, B., and Durrant-Whyte, H. F. (2002). "Information based adaptive robotic exploration," in *IEEE/RSJ international conference on intelligent robots and systems* (Lausanne, Switzerland: IROS), 540–545. doi:10.1109/IRDS.2002.1041446

Bradley, D. M., Chang, J. K., Silver, D., Powers, M., Herman, H., Rander, P., et al. (2015). "Scene understanding for a high-mobility walking robot," in *IEEE/RSJ international conference on intelligent robots and systems* (Hamburg, Germany: IROS), 1144–1151. doi:10.1109/IROS.2015.7353514

Brown, D., and Webster, G. (2010). *Now a stationary research platform, NASA's Mars rover Spirit starts a new chapter in red planet scientific studies*. Pasadena, CA: NASA Press Release.

Carrillo, H., Dames, P., Kumar, V., and Castellanos, J. A. (2018). Autonomous robotic exploration using a utility function based on Rényi's general theory of entropy. *Auton. Robots* 42, 235–256. doi:10.1007/s10514-017-9662-9

Charrow, B., Liu, S., Kumar, V., and Michael, N. (2015). Information-theoretic mapping using cauchy-schwarz quadratic mutual information. *IEEE Int. Conf. Robotics Automation (ICRA)*, 4791–4798. doi:10.1109/ICRA.2015.7139865

Dang, T., Tranzatto, M., Khattak, S., Mascarich, F., Alexis, K., and Hutter, M. (2020). Graph-based subterranean exploration path planning using aerial and legged robots. *J. Field Robot.* 37, 1363–1388. doi:10.1002/rob.21993

Faigl, J., and Čížek, P. (2019). Adaptive locomotion control of hexapod walking robot for traversing rough terrains with position feedback only. *Robotics Aut. Syst.* 116, 136–147. doi:10.1016/j.robot.2019.03.008

Faigl, J., and Kulich, M. (2013). "On determination of goal candidates in frontier-based multi-robot exploration," in *European conference on mobile robots* (Barcelona, Spain: ECMR), 210–215. doi:10.1109/ECMR.2013.6698844

Faigl, J., Kulich, M., and Přeučil, L. (2012). "Goal assignment using distance cost in multi-robot exploration," in *IEEE/RSJ international conference on intelligent robots and systems* (Vilamoura-Algarve, Portugal: IROS), 3741–3746. doi:10.1109/IROS.2012.6385660

Fankhauser, P., Bloesch, M., Gehring, C., Hutter, M., and Siegwart, R. (2014). World Scientific, 433–440.Robot-centric elevation mapping with uncertainty estimates*Mob. Serv. Robot.*

Forouhar, M., Čížek, P., and Faigl, J. (2021). "Scarab II: A small versatile six-legged walking robot," in *5th full-day workshop on legged robots at IEEE international conference on robotics and automation* (Xi'an, China: ICRA), 1–2.

Fritzke, B. (1994). "A growing neural gas network learns topologies," in *Conference on neural information processing systems* (Denver, CO: NIPS), 625–632.

Gonzalez, R., and Iagnemma, K. (2018). Slippage estimation and compensation for planetary exploration rovers. State of the art and future challenges. *J. Field Robotics* 35, 564–577. doi:10.1002/rob.21761

Guastella, D. C., and Muscato, G. (2021). Learning-based methods of perception and navigation for ground vehicles in unstructured environments: A review. *Sensors* 21, 73. doi:10.3390/s21010073

Guerrero, E., Bonin-Font, F., and Oliver, G. (2021). Adaptive visual information gathering for autonomous exploration of underwater environments. *IEEE Access* 9, 136487–136506. doi:10.1109/ACCESS.2021.3117343

Haddeler, G., Chan, J., You, Y., Verma, S., Adiwahono, A. H., and Meng Chew, C. (2020). "Explore bravely: Wheeled-legged robots traverse in unknown rough environment," in *IEEE/RSJ international conference on intelligent robots and systems* (Las Vegas, NV: IROS), 7521–7526. doi:10.1109/IROS45743.2020.9341610

Helsgaun, K. (2000). An effective implementation of the lin-kernighan traveling salesman heuristic. *Eur. J. Operational Res.* 126, 106–130. doi:10.1016/s0377-2217(99)00284-2

Hollinger, G. A., and Sukhatme, G. S. (2014). Sampling-based robotic information gathering algorithms. *Int. J. Rob. Res.* 33, 1271–1287. doi:10.1177/0278364914533443

Homberger, T., Bjelonic, M., Kottege, N., and Borges, P. V. K. (2016). "Terrain-dependant control of hexapod robots using vision," in *International symposium on experimental robotics* (Nagasaki, Japan: ISER), 92–102. doi:10.1007/978-3-319-50115-4_9

Karolj, V., Viseras, A., Merino, L., and Shutin, D. (2020). An integrated strategy for autonomous exploration of spatial processes in unknown environments. *Sensors* 20, 3663. doi:10.3390/s20133663

Kottege, N., Parkinson, C., Moghadam, P., Elfes, A., and Singh, S. P. N. (2015). Energetics-informed hexapod gait transitions across terrains. *IEEE Int. Conf. Robotics Automation (ICRA)*, 5140–5147. doi:10.1109/ICRA.2015.7139915

Krajník, T., Fentanes, J. P., Santos, J. M., and Duckett, T. (2017). Fremen: Frequency map enhancement for long-term mobile robot autonomy in changing environments. *IEEE Trans. Robot.* 33, 964–977. doi:10.1109/TRO.2017.2665664

Krüsi, P., Bosse, M., and Siegwart, R. (2016). Driving on point clouds: Motion planning, trajectory optimization, and terrain assessment in generic nonplanar environments. *J. Field Robot.* 34, 940–984. doi:10.1002/rob.21700

Lin, B., and Song, S. (1993). Dynamic modeling, stability and energy efficiency of a quadrupedal walking machine. *J. Robot. Syst.* 18, 657–670. doi:10.1002/rob.8104

Luo, W., and Sycara, K. (2018). Adaptive sampling and online learning in multi-robot sensor coverage with mixture of Gaussian processes. *IEEE Int. Conf. Robotics Automation (ICRA)*, 6359–6364. doi:10.1109/ICRA.2018.8460473

Ma, K.-C., Liu, L., Heidarsson, H. K., and Sukhatme, G. S. (2018). Data-driven learning and planning for environmental sampling. *J. Field Robot.* 35, 643–661. doi:10.1002/rob.21767

Makarenko, A. A., Williams, S. B., Bourgault, F., and Durrant-Whyte, H. F. (2002). in *IEEE/RSJ international conference on intelligent robots and systems*, 1, An experiment in integrated exploration534–539. doi:10.1109/IRDS.2002.1041445(IROS)

Mann, H. B., and Whitney, D. R. (1947). On a test of whether one of two random variables is stochastically larger than the other. *Ann. Math. Stat.* 18, 50–60. doi:10.1214/aoms/1177730491

Martin, S., and Corke, P. (2014). Long-term exploration & tours for energy constrained robots with online proprioceptive traversability estimation. *IEEE Int. Conf. Robotics Automation (ICRA)*, 5778–5785. doi:10.1109/ICRA.2014.6907708

Mayuku, O., Surgenor, B. W., and Marshall, J. A. (2021). "A self-supervised near-to-far approach for terrain-adaptive off-road autonomous driving," in *IEEE international conference on robotics and automation* (Xi'an, China: ICRA), 14054–14060. doi:10.1109/ICRA48506.2021.9562029

McGhee, R. B., and Frank, A. A. (1968). On the stability properties of quadruped creeping gaits. *Math. Biosci.* 3, 331–351. doi:10.1016/0025-5564(68)90090-4

Moravec, H., and Elfes, A. (1985). "High resolution maps from wide angle sonar," in *1985 IEEE international conference on robotics and automation proceedings*, 116–121. doi:10.1109/ROBOT.1985.1087316

Noon, C. E., and Bean, J. C. (1993). An efficient transformation of the generalized traveling salesman problem. *INFOR Inf. Syst. Operational Res.* 31, 39–44. doi:10.1080/03155986.1993.11732212

Noon, C. E. (1988). *The generalized traveling salesman problem*. Ann Arbor, MI: Ph.D. thesis, University of Michigan.

O'Callaghan, S., Ramos, F. T., and Durrant-Whyte, H. (2009). Contextual occupancy maps using Gaussian processes. *IEEE Int. Conf. Robotics Automation (ICRA)*, 1054–1060. doi:10.1109/ROBOT.2009.5152754

O'Meadhra, C., Tabib, W., and Michael, N. (2019). Variable resolution occupancy mapping using Gaussian mixture models. *IEEE Robot. Autom. Lett.* 4, 2015–2022. doi:10.1109/LRA.2018.2889348

Ossenkopf, M., Castro, G., Pessacg, F., Geihs, K., and De Cristóforis, P. (2019). "Long-Horizon Active SLAM system for multi-agent coordinated exploration," in *European conference on mobile robots* (Prague, Czech Republic: ECMR), 1–6. doi:10.1109/ECMR.2019.8870952

Papadakis, P. (2013). Terrain traversability analysis methods for unmanned ground vehicles: A survey. *Eng. Appl. Artif. Intell.* 26, 1373–1385. doi:10.1016/j.engappai.2013.01.006

Pasolli, E., and Melgani, F. (2011). Gaussian process regression within an active learning scheme. *IEEE Int. Geoscience Remote Sens. Symposium*, 3574–3577. doi:10.1109/IGARSS.2011.6049994

Prágr, M., Čížek, P., Bayer, J., and Faigl, J. (2019a). "Online incremental learning of the terrain traversal cost in autonomous exploration," in *Robotics: Science and systems, (RSS)* (Freiburg im Breisgau, Germany). 1–10. doi:10.15607/RSS.2019.XV.040

Prágr, M., Čížek, P., and Faigl, J. (2018). "Cost of transport estimation for legged robot based on terrain features inference from aerial scan," in *IEEE/RSJ international conference on intelligent robots and systems (IROS)* (Prague, Czech Republic: IEEE), 1745–1750. doi:10.1109/IROS.2018.8593374

Prágr, M., Čížek, P., and Faigl, J. (2019b). "Incremental learning of traversability cost for aerial reconnaissance support to ground units," in *2018 modelling and simulation for autonomous systems* (Prague, Czech Republic: MESAS), 412–421. doi:10.1007/978-3-030-14984-0_30

Prudent, Y., and Ennaji, A. (2005). An incremental growing neural gas learns topologies. *Int. Jt. Conf. Neural Netw. (IJCNN)* 2, 1211–1216. doi:10.1109/IJCNN.2005.1556026

Quann, M., Ojeda, L., Smith, W., Rizzo, D., Castanier, M., and Barton, K. (2020). Off-road ground robot path energy cost prediction through probabilistic spatial mapping. *J. Field Robot.* 37, 421–439. doi:10.1002/rob.21927

Quigley, M., Conley, K., Gerkey, B. P., Faust, J., Foote, T., Leibs, J., et al. (2009). *ICRA Workshop on Open Source Software*, 1–6.Ros: An open-source robot operating system.

Ramos, F., and Ott, L. (2016). Hilbert maps: Scalable continuous occupancy mapping with stochastic gradient descent. *Int. J. Rob. Res.* 35, 1717–1730. doi:10.1177/0278364916684382

Rasmussen, C. E., and Williams, C. K. I. (2006). Gaussian processes for machine learning. *Adaptive computation and machine learning*. Cambridge, Mass: MIT Press.

Rényi, A. (1961). On measures of entropy and information. *Berkeley Symposium Math. Statistics Probab.*, 547–561.

Rhodes, C., Liu, C., and Chen, W.-H. (2020). "Informative path planning for gas distribution mapping in cluttered environments," in *IEEE/RSJ international conference on intelligent robots and systems* (Las Vegas, NV: IROS), 6726–6732. doi:10.1109/IROS45743.2020.9341781

Schultz, A. C., Adams, W., and Yamauchi, B. (1999). Integrating exploration, localization, navigation and planning with a common representation. *Auton. Robots* 6, 293–308. doi:10.1023/A:1008936413435

Shi, Y., Wang, N., Zheng, J., Zhang, Y., Yi, S., Luo, W., et al. (2020). "Adaptive informative sampling with environment partitioning for heterogeneous multi-robot systems," in *IEEE/RSJ international conference on intelligent robots and systems* (Las Vegas, NV: IROS), 11718–11723. doi:10.1109/IROS45743.2020.9341711

Singh, A., Krause, A., Guestrin, C., Kaiser, W., and Batalin, M. (2007). "Efficient planning of informative paths for multiple robots," in *International joint conference on artifical intelligence*, 2204–2211.

Sofman, B., Lin, E., Bagnell, J. A., Cole, J., Vandapel, N., and Stentz, A. (2006). Improving robot navigation through self-supervised online learning. *J. Field Robot.* 23, 1059–1075. doi:10.1002/rob.20169

Srinivas, N., Krause, A., Kakade, S., and Seeger, M. (2010). "Gaussian process optimization in the bandit setting: No regret and experimental design," in *Intl. Conf. International conference on machine learning (ICML)* (Haifa, Israel), 1015–1022.

Stachniss, C., Grisetti, G., and Burgard, W. (2005). "Information gain-based exploration with rao-blackwellized particle filters," in *Robotics: Science and systems*, 1–8. doi:10.15607/RSS.2005.I.009

Stelzer, A., Hirschmüller, H., and Görner, M. (2012). Stereo-vision-based navigation of a six-legged walking robot in unknown rough terrain. *Int. J. Rob. Res.* 31, 381–402. doi:10.1177/0278364911435161

Vallvé, J., and Andrade-Cetto, J. (2015). Potential information fields for mobile robot exploration. *Robotics Aut. Syst.* 69, 68–79. doi:10.1016/j.robot.2014.08.009

Viseras, A., Shutin, D., and Merino, L. (2019). Robotic active information gathering for spatial field reconstruction with rapidly-exploring random trees and online learning of Gaussian processes. *Sensors* 19, 1016. doi:10.3390/s19051016

Wang, J., and Englot, B. (2016). Fast, accurate Gaussian process occupancy maps via test-data octrees and nested Bayesian fusion. *IEEE Int. Conf. Robotics Automation (ICRA)*, 1003–1010. doi:10.1109/ICRA.2016.7487232

Wermelinger, M., Fankhauser, P., Diethelm, R., Krüsi, P., Siegwart, R., and Hutter, M. (2016). "Navigation planning for legged robots in challenging terrain," in *IEEE/RSJ international conference on intelligent robots and systems*, 1184–1189. doi:10.1109/IROS.2016.7759199

Yamauchi, B. (1997). A frontier-based approach for autonomous exploration. *CIRA (IEEE)*, 146–151. doi:10.1109/CIRA.1997.613851

Zlot, R., and Stentz, A. (2006). Market-based multirobot coordination for complex tasks. *Int. J. Rob. Res.* 25, 73–101. doi:10.1177/0278364906061160

Check for updates

Autonomous exploration with online learning of traversable yet visually rigid obstacles

Miloš Prágr[1] · **Jan Bayer**[1] · **Jan Faigl**[1]

Abstract
This paper concerns online learning of terrain properties combining haptic perception with exteroceptive sensing to reason about forces needed to pass through terrains that visually appear as untraversable obstacles. Terrain learning is studied within the context of autonomous exploration. We propose predicting the traversability of potentially obstructing terrains by active perception to establish a connection between the observed geometric environment model and deliberately sampled forces to pass through the terrain using a haptic sensor that probes the terrain in front of the robot. The developed solution uses a Gaussian Process regressor in online learning and force prediction. The robot is navigated by following the information gain to improve traversability and spatial models. The proposed approach has been experimentally verified in fully autonomous exploration with a multi-legged walking robot. The robot is navigated through visually looking obstacles and explores "hidden" areas while following the expected information gain to explore the terrain properties of the mission area.

Keywords Mobile robot · Exploration · Active perception · Haptic sensing · Gaussian Process regression

1 Introduction

Terrain properties like appearance and geometry can be used to reason about the traversability of mobile robots by assigning terrain classes (Bradley et al., 2015), computing a terrain traversal cost function (Sofman et al., 2006), or discriminating untraversable terrains (Stelzer et al., 2012). Further, we reason about terrains that appear untraversable due to their geometrical properties. These visually appearing obstacles can be traversable, such as sparse vegetation or a curtain-covered doorway, which appear as a wall when presumed to be rigid. Assuming a non-rigid terrain is a rigid obstacle to

avoid leads to safe behavior. However, in autonomous exploration, such behavior might be overly cautious and results in an incomplete terrain model if the robot is prevented from visiting areas separated by terrain that appears untraversable yet can be traversed by the robot.

In this paper, we propose to use both exteroceptive and haptic sensing to actively learn to predict the traversability of potentially obstructing obstacles. A haptic sensor (3D-printed bumper) is used to sample the force needed to pass through these visually appearing obstacles. A Gaussian Process (GP) regressor (Rasmussen & Williams, 2006) exploits the obstacle appearance in online learning and predicts the forces required to be generated by the robot to pass through an obstacle. The proposed method is demonstrated in a real-world autonomous exploration scenario where a multi-legged walking robot actively learns terrains that can be passed through. Besides exploring the unobserved areas, the robot actively collects information about the force prediction model driven by the expected information gain from interaction with potential obstacles. An example of the robot decision-making using the proposed model is visualized in Fig. 1.

Regarding the existing work, including our previous work on terrain learning (Prágr et al., 2018b, 2019), the main contributions of the presented work are considered as follows.

Miloš Prágr
pragrmi1@fel.cvut.cz

Jan Bayer
bayerja1@fel.cvut.cz

Jan Faigl
faiglj@fel.cvut.cz

[1] Department of Computer Science, Faculty of Electrical Engineering, Czech Technical University in Prague, Prague, Czech Republic
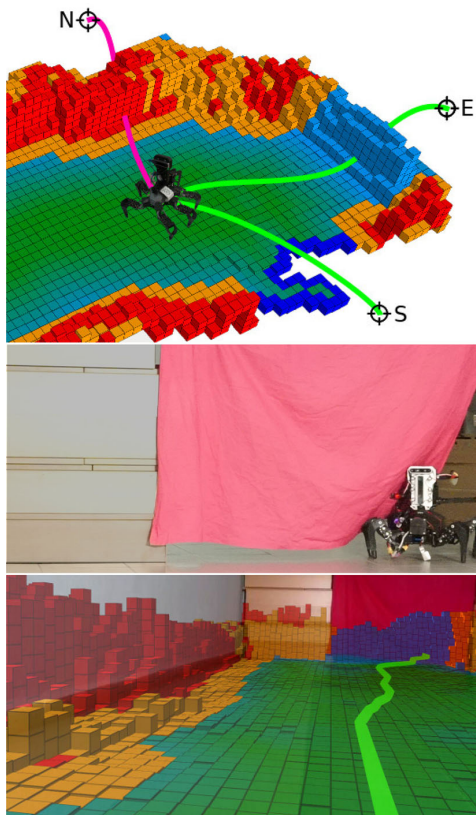
Fig. 1 A visualization of autonomous decision-making in the proposed terrain learning approach; untraversable terrains are in the red and orange, while the green and blue areas can be traversed. For the "blue obstacles", the robot has already learned that such visually appearing obstacles can be traversed through. (top) The circular black targets represent possible exploration goal locations. The southern (S) goal is reachable over terrain that appears traversable, while both the northern (N) and eastern (E) goals are located behind obstacles. However, the eastern goal is considered reachable, visualized by the green path from the current robot's location to the possible goal location. (middle) Robot walking through a traversable obstacle in the form of purple fabric. (bottom) Projection of the traversability on robot vision a few moments before traversing the fabric. Since the robot has learned that the purple fabric (shown at the northeast of the view) is traversable, the robot plans a path through it (Color figure online)

– Model characterizing the force needed to pass through terrains (obstacles), incrementally learnable using exteroceptive and haptic measurements.
– Proposed model's capability to exploit observations about terrain traversability that visually appears to be rigid, but for which the experienced haptic interaction provides evidence of its traversability.
– A robotic system with active haptic perception using information-theoretic estimation of the expected information gain of the object touching to ensure traversability of terrains that visually appear impassable.

– Experimental evaluation of the developed incremental model learning and online prediction in an autonomous robotic exploration scenario with a real hexapod walking robot. The model's incremental learning capacity is exploited in online decision-making based on the expected information gain from visiting and interacting with terrain obstacles.
– Experimental evaluation of the developed incremental model learning with real outdoor vegetation.

The rest of the paper is structured as follows. Section 2 provides a brief review of the most related approaches that concern terrain traversal by unmanned ground vehicles and mobile robot exploration. Section 3 specifies the problem of mobile robot exploration of the environment with terrains that appear untraversable yet can be traversed. The proposed learning and modeling of such environments and the autonomous exploration framework to build a complete traversability map of the unknown environment with areas "hidden" by non-rigid terrains visually appearing as obstacles are proposed in Sect. 4. The employed haptic sensor is overviewed in Sect. 5, together with the reported evaluation results from the real experimental deployment of the proposed system. Section 6 concludes the paper.

2 Related work

In this section, a short survey of the related work is provided. First, we focus on approaches concerned with describing terrain traversability, and the herein presented work is placed in their context. Second, we provide a short overview of robotic exploration.

2.1 Terrain traversability

It is not desirable to enter areas that would bar mobile robots from continuing their missions by damaging the robot, such as by falling off a high cliff or impeding further motion as it happened by getting stuck in soft sand (Brown & Webster, 2010). Besides, avoiding terrains that do not pose an immediate danger to the robot but are hard to traverse is desirable. Such terrains may cause unnecessary energy consumption or slowly wear the robot body. Thus, autonomous mobile robots have to consider local terrain properties during navigation through the operational environment. A rich body of literature reports on systems concerning terrain traversability. An extensive review and taxonomy of such approaches can be found in Papadakis (2013).

The terrain traversability can be described either by classifying the terrains into a set of terrain classes (Belter et al., 2019; Giguere & Dudek, 2008; Kragh et al., 2015; Rothrock et al., 2016), or by assigning a continuous traversability score

to the observed terrain properties (Kottege et al., 2015), such as terrain appearance and geometry (Prágr et al., 2018a; Prágr & Faigl, 2019). A common yet straightforward approach is to classify terrain either as an untraversable obstacle or a free traversable space. Stelzer et al. (2012) use geometric terrain properties to classify terrain as untraversable or free and compute a continuous index to describe the traversability of the latter class. In Kragh et al. (2015), the Support Vector Machine (SVM) classifier learns three classes (ground, vegetation, and object) in an agricultural environment; while 40 hand-labeled terrain classes are used in Bradley et al. (2015), where some of them are denoted as obstacles.

Terrain traversability scores are computed directly from remotely observed terrain appearance and geometry or describe the difficulty or energy consumption previously experienced by the robot when traversing over the respective terrain. Sofman et al. (2006) use overhead imagery to learn traversability log-scale score based on ground LiDAR data. Overhead features are utilized to predict the energy required to traverse various outdoor terrains in Prágr et al. (2020). The cost of transport, an energy-over-velocity cost originating in biology (Tucker, 1975), is modified for use with battery-powered robots in Kottege et al. (2015). McGhee and Frank (1968) propose to measure the stability of a multi-legged robot in terms of its foothold support polygon. Furthermore, Prágr et al. (2019) learn to predict stability based on inertial measurements of the robot shaking in an active perception scenario.

Terrain geometry, which can serve as a traversability indicator, can be characterized in terms of its slope (Brunner et al., 2013; Gu et al., 2008), step height (Homberger et al., 2016), or roughness (Belter et al., 2019; Krüsi et al., 2016). The terrain shape is described based on the Eigen-statistics of the point cloud covariance matrix in Lalonde et al. (2006) and Kragh et al. (2015). Approaches that consider terrain color use the HSV (Sofman et al., 2006) or Lab (Otsu et al., 2016) color space to avoid illumination sensitivity of the RGB color space. Cunningham et al. (2019) propose to use thermal imagery to predict slip during Mars rover missions.

Autonomous robots operating in outdoor environments might encounter hard-to-traverse-vegetation, and thus attention is given to such terrains (Bradley et al., 2015; Sofman et al., 2006). Ünsalan and Boyer (2004) compared indices that characterize vegetation using a LiDAR sensor. In Petrou et al. (2015), the vegetation height is classified using overhead imagery. The elevation of the supporting terrain occluded by the vegetation is estimated using a GP model of the vegetation height and foothold supports in Homberger et al. (2019).

Furthermore, mobile robots may also be deployed in environments with dynamic obstacles such as closing doors or moving people. Approaches to handle such dynamic environments may either filter out dynamic objects and extract a static map (Burgard et al., 1999), or use spatial-temporal maps to represent and predict changes in the environment such as a door being closed or opened (Biber & Duckett, 2005; Halodová et al., 2019). However, to the best of the authors' knowledge, none of the existing dynamic environment approaches is designed to handle terrain that appears like a non-moving obstacle for the whole time while it can be passed through with sufficient forward force.

In Baleia et al. (2015), a haptic antenna is used to classify the traversability of visually untraversable yet possibly traversable objects. The therein proposed traversability predictor is used in a self-supervised manner. Upon encountering a potential obstacle during its mission execution, the robot recalls the *k*-nearest appearing obstacles using a feature similarity metric. The robot computes its confidence levels regarding the obstacle being traversable and untraversable and decides whether to move forward or avoid the obstacle. If neither the obstacle traversability nor non-traversability can be observed with sufficient confidence, the robot uses the antenna to assess the obstacle and expand its memory.

Kahn et al. (2021) present a self-supervised, end-to-end learning system to navigate potentially traversable terrains that appear untraversable without relying on a Simultaneous Localization and Mapping (SLAM) system. The robot uses a random walk policy to collect a dataset, where it identifies collision, bumpiness, and position events using its Inertial Measurement Unit (IMU) and wheel odometry. The robot learns to predict the events given input image and action. The learned models can be exploited in navigation with respect to (w.r.t.) an arbitrary reward function that considers the three event types.

In the herein presented work, we aim to build both the observed geometric model of the environment and a model that predicts the traversability of potentially obstructing terrains by a small amount of data provided by a haptic sensor correlated with exteroceptive sensing. Compared to Baleia et al. (2015) and Kahn et al. (2021), we characterize the force needed to pass through potential obstacles, and hence the proposed approach is robot agnostic. Unlike the method proposed by Baleia et al. (2015), which focuses on classifying objects encountered during the robot mission that can be avoided when deemed untraversable, we address relatively large, obstacle-like terrains that might block access to additional sections of the environment.

Besides, considering the previous work on learning using collected data (Kahn et al., 2021), the proposed system is employed in autonomous exploration with the additional complexity of incremental learning of the force to pass through obstacles on the robot during the deployment. Based on the visual appearance, the force predictions are utilized in online decision-making to discriminate the objects the robot cannot traverse and identify unknown obstacles the robot should sample next. Since our approach is focused on learning the force, it samples and learns only on terrains that are

unknown and thus informative. We use a learning method that requires tens of samples and can be used online, directly on the robot during the exploration. Thus, the proposed approach uses much less data than general approaches such as Kahn et al. (2021), which rely on long-term data collection and offline processing.

Because we employ the proposed approach in the exploration context, an overview of mobile robot exploration approaches is presented in the following paragraphs.

2.2 Mobile robot exploration

Mobile robot exploration is an active perception scenario where one or a group of mobile robots build a model of the mission environment. In frontier-based exploration (Faigl & Kulich, 2015), the robot follows frontiers, the boundaries between the observed traversable and not yet observed areas (Yamauchi, 1997). Alternatively, the probabilistic representation of the cell occupancy in the occupancy grids (Moravec & Elfes, 1985) can be used in the exploration strategy that maximizes the information gain (Bourgault et al., 2002; Makarenko et al., 2002). Beside grid maps, Gaussian Processes (GPs) (Ruiz & Olariu, 2015; Vasudevan et al., 2009), Gaussian Mixture Models (GMMs) (O'Meadhra et al., 2019), or Hilbert maps (Ramos & Ott, 2016) can be used to create continuous maps that are not resolution-dependant. Since the GP regressors provide predictive variance for their queries, they are particularly suited for active perception scenarios. Jadidi et al. (2018) use a GP-based representation to construct frontier maps, while the GMM is used in Tabib et al. (2019).

Exploration is not limited to building maps and geometric models but may also concern modeling a phenomenon underlying the spatial environment such as temperature (Luo & Sycara, 2018). In informative path planning (Singh et al., 2007), the goal is to find the most informative path subject to a particular constraint, such as the robot energy budget. Hence, the robot explores as much of the environment as possible while avoiding battery depletion that would lead to its immobilization, as noted in Tiwari et al. (2019), where a framework for operation range estimation is presented to support robots ranging from multi-rotor fliers to ground vehicles.

When the goal is to find extrema of the modeled phenomena, exploration-exploitation tradeoff-based approaches such as Gaussian Process Upper Confidence Bound (Srinivas et al., 2010) can be utilized. Furthermore, the active learning of the underlying model can be combined with the traditional geometric exploration (Prágr et al., 2019). For example, the robot localization model can be incorporated into information-based exploration approaches, such as the localization uncertainty represented using the differential entropy of the robot position distribution by directly adding

it to the mapping uncertainty (Bourgault et al., 2002; Stachniss et al., 2005). However, since the differential entropy differs from the Shannon entropy of the binary cell occupancy distribution in scale, particularly when considering dynamic environment size, Carrillo et al. (2018) argue that it is not desirable to combine them directly and employ Rényi entropy (Rényi, 1961) to create an uncertainty utility function.

In this paper, we present a combination of the spatial map exploration with the active building of the obstacle traversability model characterized as the force to pass through. The proposed approach is demonstrated within autonomous robotic exploration, in an escape-like scenario, where the robot first explores all areas accessible without interacting with obstacles. Only when no such areas are available does it actively learn the obstacle traversability. Consequently, the robot selects the exploration goals independently for the respective models. Hence, even though the spatial and prediction models yield information gains in Shannon's discrete and differential entropy, respectively, we circumvent the need to combine these two quantities.

3 Problem specification

We address mobile robot exploration in environments where obstacles can be non-rigid and passable by the robot. The robot is tasked to explore an environment modeled as the grid map $\mathcal{M}_{2.5D}$, where each cell v corresponds to a foothold of the hexapod walking robot used in the experimental verification. Hence, the cell size d_v corresponds to the foothold size. The robot moves through the environment along a path ψ that can be expressed as

$$
\begin{aligned}
&\psi = (v_1, v_2, \ldots, v_n), \\
&\text{s.t.} \\
&\forall i \in 1, \ldots, n : \ p(v_i) = 1, \\
&\forall i \in 1, \ldots, n-1 : \ v_{i+1} \in 8nb(v_i),
\end{aligned}
\tag{1}
$$

where $8nb(v)$ is the 8-neighborhood function on the grid, and $p(v)$ returns the probability that the cell v is passable, denoted as the robot's traversability.

Flat areas are considered traversable, and the environment geometry is used to determine areas that appear as obstacles. Since obstacles may be non-rigid, the robot's traversability through such areas cannot be determined only by geometry. Rather, an obstacle is traversable if the robot can exert force sufficient to pass through. Hence, the robot's traversability through a grid cell v that appears as an obstacle is

$$
p(v) = \begin{cases} 0 & \text{if } F(v) > \mathrm{F_{trav}}, \\ 1 & \text{otherwise,} \end{cases}
\tag{2}
$$

where $F(v)$ is the force needed to pass through the cell v, and the threshold F_{trav} is the maximum force that can be exerted by the robot when trying to pass through the obstacle.

In the explored environment, the force to pass through the obstacles is not known for the individual obstacles. However, it is assumed the force is similar for similar-appearing obstacles, and thus the robot can predict the force needed to pass through the obstacles described by their respective appearance descriptors A as

$$f_{predict} : A \rightarrow \hat{F}. \tag{3}$$

Besides, while we assume that the appearance description is sufficiently discriminative to distinguish the obstacles in each individual explored environment, similar-appearing obstacles in different deployments may have different rigidity. For example, dry summer grass is easier to traverse than wet grass prevalent during spring, even though they appear similar. Hence, the robot learns the rigidity predictor $f_{predict}$ online during the exploration, starting from scratch for each deployment.

Since the task of the robot is to explore the environment where some areas may be reached only by traversing through the non-rigid obstacles, the portion of the environment that is explored is the benchmark value. The proposed method is thus evaluated and compared to a baseline model that considers the obstacles untraversable.

4 Proposed traversability model

The proposed method to characterize the traversability of apparent yet potentially traversable obstacles is presented as a part of the autonomous exploration. The robot is equipped with a haptic bumper sensor to experience the possible traversability of particular terrain areas. The traversability of obstructing obstacles the robot can walk through is characterized by sampling the force needed to pass through the obstacle. These haptic measurements are considered the traversability ground truth. The robot incrementally learns a Gaussian Process (GP) regressor (Rasmussen & Williams, 2006) employed to predict the force to pass through from the appearance of the apparent obstacles and thus to assess the traversability of the obstacles.

The idea of the proposed traversability model is demonstrated in an exploration-exploitation scenario set up as a robotic *escape mission*. The robot first explores the areas observed by the exteroception that appear traversable without interacting with apparent obstacles. After all such reachable areas are explored, the robot actively uses its haptic sensor to learn a model of obstacle traversability (force model exploration). Furthermore, when the robot learns that some apparent obstacle is traversable, it reverts to exploring the

area that may lie behind such obstacles (force model exploitation).

The relation of the proposed terrain model, its learning based on the measured sensory input, and the decision-making in the exploration setup is depicted in Fig. 2. It principally works as follows. The *exteroceptive* part is responsible for continuously building an elevation map of the robot's surroundings using RGB-D sensory input. Further, the *exteroceptive* model identifies the areas that appear untraversable by the robot from terrain geometry. The geometric properties of the terrain are then passed to the *learning* module. The *haptic* module accumulates measurements of the force needed to pass through the apparent obstacles as the force experienced by the haptic sensor during the mission. These ground-truth force measurements are also passed to the *learning* module, which pairs the respective appearance characterization from the *exteroceptive* model with these experienced haptic observations. The module realizes online learning and prediction of the force, and thus the traversability for the observed yet untraversed terrains. The *exteroceptive* model is updated with these traversability predictions to allow traversal through non-rigid obstacles. In *exploration*, the robot uses information gain predictions provided by the *exteroceptive* and *learning* parts to select the next exploration goal.

The cell traversability $\hat{p}(v)$ is reported based on the inputs from the *haptic*, *exteroceptive*, and *learning* modules. The cells with traversability measurements $p_{haptic}(v)$ are considered ground truth and reported as $\hat{p}(v) = p_{haptic}(v)$, regardless of the other modules. If there is no ground truth traversability reported for the cell by the *haptic* model, its geometric traversability $p_{geom}(v)$ provided by the *exteroceptive* module is used. If such cell appears traversable, the traversability is set to $\hat{p}(v) = p_{geom}(v) = 1$. For cells marked as potential obstacles with $p_{geom}(v) = 0$, the traversability is assessed by the *learning* module, reporting the traversability prediction $p_{predict}(v)$. Finally, unobserved cells are reported as traversable $\hat{p}(v) = p_{unobserved} = 1$ to allow traversal of such areas, which is desirable since cells hidden directly behind traversable obstacles cannot be observed before traversal. The traversability assesment can be summarized as

$$\hat{p}(v) = \begin{cases} p_{haptic}(v) & \text{if } p_{haptic}(v) \text{ is known} \\ p_{predict}(v, \mathcal{P}) & \text{if } p_{geom}(v) \text{ is known and } p_{geom}(v) = 0 \\ 1 & \text{otherwise} \end{cases} \tag{4}$$

The individual exteroceptive and haptic models, traversability predictions, and the terrain learning process are detailed in the following sections. The symbols used in the description are overviewed in Table 1.
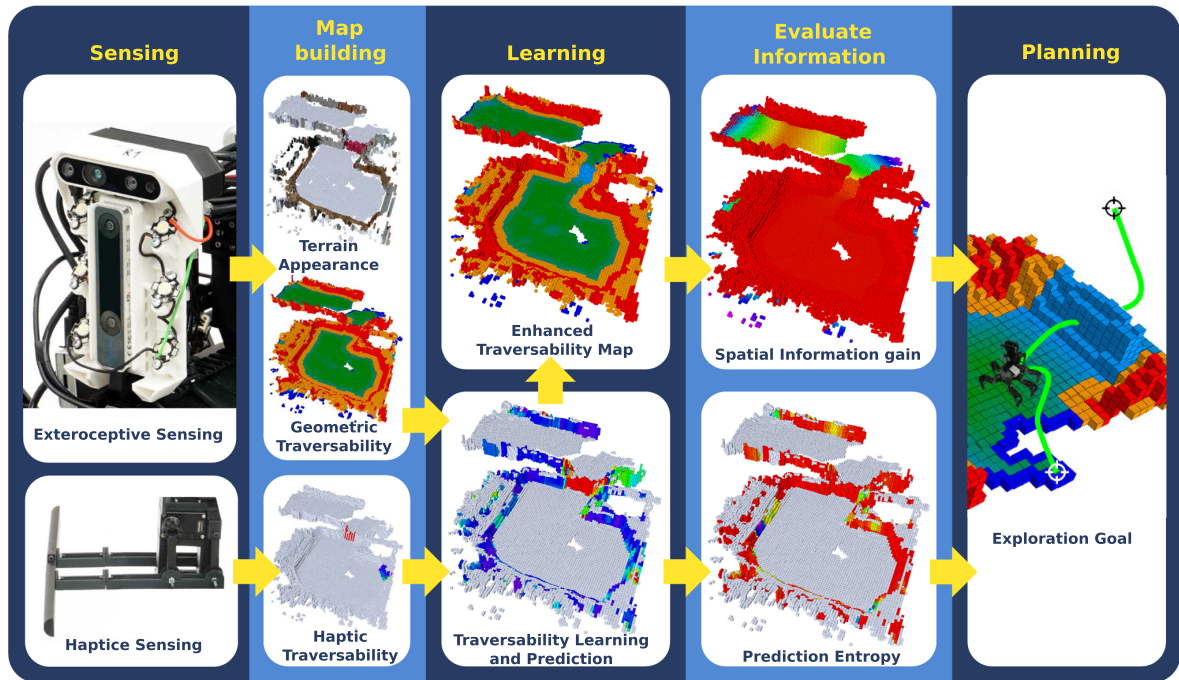
Fig. 2 Individual modules employed in building the traversability model and its usage in the autonomous exploration

4.1 Exteroceptive model

Robot's visual and depth perceptions are utilized to construct a colored elevation map $\mathcal{M}_{2.5D}$, see Fig. 3a. The elevation map is a grid map with the squared cell of the size d_ν, and its underlying representation is based on a memory-efficient quadtree data structure (Bayer & Faigl, 2020). For each cell $\nu \in \mathcal{M}_{2.5D}$, the geometrical traversability model (visualized in Fig. 3b) provides $p_{geom}(\nu)$, the probability that the robot can traverse the cell ν, by comparing local differences of height to the threshold t_{trav}

$$p_{geom}(\nu) \begin{cases} 0 & \text{if } \max_{\xi \in 8nb(\nu)} \Delta(\nu, \xi) > t_{trav} \\ 1 & \text{otherwise} \end{cases}, \quad (5)$$

where $8nb(\nu)$ is the 8-neighborhood of the cell ν, the particular value of t_{trav} depends on the kinematics of the used robot. The step height $\Delta(\nu_a, \nu_b)$ is defined as

$$\Delta(\nu_a, \nu_b) = |elevation(\nu_a) - elevation(\nu_b)|, \quad (6)$$

with $elevation(\nu)$ denoting the estimated height of the terrain at the cell ν. Note that all the results presented in this paper are for $t_{trav} = 12$cm based on the kinematics constraints and motion gait of the utilized hexapod walking robot.

The information about the geometric traversability model gained by observing an unknown cell ξ is approximated as the entropy of the binary distribution p_{geom} that depends on the 8-neighborhood of the cell. Since the knowledge whether one cell is traversable corresponds to one bit, the information gained by observing ξ with unknown height is approximated as

$$I_{geom}^{cell}(\xi) \approx \frac{k(\xi) + 1}{9}, \quad (7)$$

where $k(\xi)$ is the number of the unknown cells in the neighborhood of ξ. Thus, the expected information gained by perceiving the terrain from the position of the cell ν is

$$I_{geom}^{model}(\nu) = \sum_{\xi \in \delta(\nu, \delta_{sensor})} \begin{cases} I_{geom}^{cell}(\xi) & \text{if } observable(\nu, \xi) \\ 0 & \text{otherwise} \end{cases}, \quad (8)$$

where $\delta(\nu, \delta_{sensor})$ is the neighborhood of ν defined by the sensor range δ_{sensor}, which value depends on the used sensor, e.g., $\delta_{sensor} = 2$ m. The function $observable(\nu, \xi)$ returns true if the cell ξ is observable from ν, which is determined by casting a ray from ν to ξ in the current elevation map $\mathcal{M}_{2.5D}$. An example of the information gain of the geometric traversability model is depicted in Fig. 3c.

Table 1 Used symbols

Description	Symbol	Description	Symbol
First introduced in Sect. 3			
Environment gridmap	$\mathcal{M}_{2.5D}$	Gridmap cell	ν
Cell size	d_ν	Path	ψ
Cell traversability	p	Cell grid 8-neighborhood	$8nb$
Force to pass through	F	Maximum force exerted by robot	F_{trav}
Terrain appearance	A	Force prediction function	$f_{predict}$
First introduced in Sect. 4			
Reported traversability	$\hat{p}$	Ground-truth haptic traversability	p_{haptic}
Apparent geometric prediction	p_{geom}	Predicted traversability	$p_{predict}$
Unobserved-cell traversability	$p_{unobserved}$		
First introduced in Sect. 4.1			
Step height	Δ	Maximum allowed step height	t_{trav}
Information gained by observing a cell	I_{geom}^{cell}	Information gained by observing from a cell	I_{geom}^{model}
Cell δ neighborhood	δ	Sensor range	δ_{sensor}
Spatial goal cluster radius	d_{cl}	Spatial goal minimum cluster size	n_{cl}^{thr}
Spatial goal set	$\mathcal{G}_{geom}$		
First introduced in Sect. 4.2			
Force measurement	z^{force}	k-th force measurement at cell	z_k^{force}
Force reported at cell after k measurements	F_k	Force uncertainty at cell after k measurements	σ_k^2
Bumper sensor measurement uncertainty	σ_{sensor}^2		
First introduced in Sect. 4.3			
Traversability predictor	$\mathcal{P}$	Mono-color appearance descriptor	A^{AB}
Force prediction mean	μ_F	Force prediction variance	σ_F^2
Approximated predictor information gain	$I_{predict}$		
First introduced in Sect. 4.4			
GP model noise variance	σ_n^2	Exponential kernel output variance	σ_{exp}^2
Exponential kernel lengthscale	l_{exp}		
First introduced in Sect. 4.5			
Planning cost	c	The cheapest path	ψ^*
Exploration goal	$\nu_{explore}^*$	Spatial exploration goal	ν_{geom}^*
Predictor exploration goal	$\nu_{predict}^*$	Predictor initialization goal	$\nu_{predict}^{*-\Psi}$
Predictor learning goal	$\nu_{predict}^{*-I}$	Minimum prediction goal information	$I_{predict}^{thr}$
First introduced in Sect. 5.3			
Histogram appearance descriptor	A^{hist}	Histogram descriptor radius	r_{hist}

Some single-use symbols are omitted for simplicity

$$0 \quad\quad 1200 \quad\quad 2400 \quad\quad 3600$$
Information Gain of Geometric Model [bit]

(a)   **(b)**   **(c)**
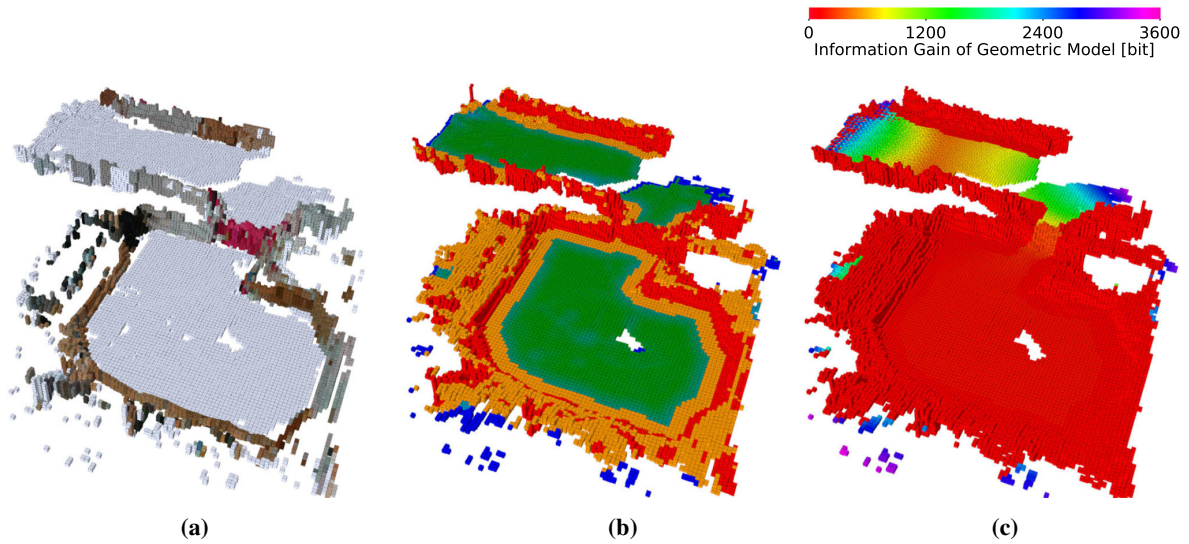
Fig. 3 Example of **a** the colored elevation grid map (obstacles colored, the ground is in the light blue); **b** the potential obstacles determined solely on their geometric properties (obstacles are in the red, traversable terrain in the green, and cells too close to obstacles in the yellow); and **c** the information gain of the geometric model indicating terrain areas with unsure spatial traversability because of lack of strong evidence from exteroceptive measurements (Color figure online)

In an active perception scenario, the goal locations $\mathcal{G}_{geom}$ to improve the geometric traversability model are selected as a subset of cells with non-zero information gain according to Algorithm 1.

4.2 Haptic model

The *haptic* model uses measurements provided by the bumper sensor to predict the traversability of the observed obstacles that can be non-rigid. In particular, the sensor measures the force $F(\nu)$ needed to pass through the obstacle at

---

Algorithm 1: Goal locations clustering.

Input: $\mathcal{M}_{2.5D}$ – 2.5D grid map with assessed frontiers,
$\quad\quad$ d_{cl} – Cluster radius, n_{cl}^{thr} – Minimal cluster size.
Output: $\mathcal{G}_{geom}$ – Clustered cells with non-zero entropy.

1 $\quad A \leftarrow \emptyset$ $\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad$ ▷ Init. set of clusters.
2 $\quad$ **for** $\nu \in \mathcal{M}_{2.5D} : I_{geom}^{model}(\nu) > 0$ **do** $\quad\quad\quad\quad\quad\quad\quad\quad$ ▷ For each map cell with non-zero entropy.
3 $\quad\quad$ **if** $A = \emptyset$ **then** $\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad$ ▷ If no clusters in set.
4 $\quad\quad\quad$ $A \leftarrow \{\{\nu\}\}$ $\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad$ ▷ Create a new cluster.
5 $\quad\quad$ **else**
6 $\quad\quad\quad$ $d \leftarrow$ `distanceToClosestCluster`(ν, A)
7 $\quad\quad\quad$ **if** $d < d_{cl}$ **then**
8 $\quad\quad\quad\quad$ `addToClosestCluster`(ν, A) $\quad\quad\quad\quad\quad\quad\quad$ ▷ Add point to existing cluster.
9 $\quad\quad\quad$ **else**
10 $\quad\quad\quad\quad$ $A \leftarrow A \cup \{\{\nu\}\}$ $\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad$ ▷ Create new cluster.
11
12
13 $\quad \mathcal{G}_{geom} \leftarrow \emptyset$ $\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad$ ▷ Init. cluster representants.
14 $\quad$ **for** $A_i \in A$ **do** $\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad$ ▷ For each clusters.
15 $\quad\quad$ **if** $|A_i| > n_{cl}^{thr}$ **then**
16 $\quad\quad\quad$ $\mathcal{G}_{geom} = \mathcal{G}_{geom} \cup \{$`cellClosestToAverageCoordinates`$(A_i)\}$ $\quad$ ▷ Create new representants.
17 $\quad$ **return** $\mathcal{G}_{geom}$
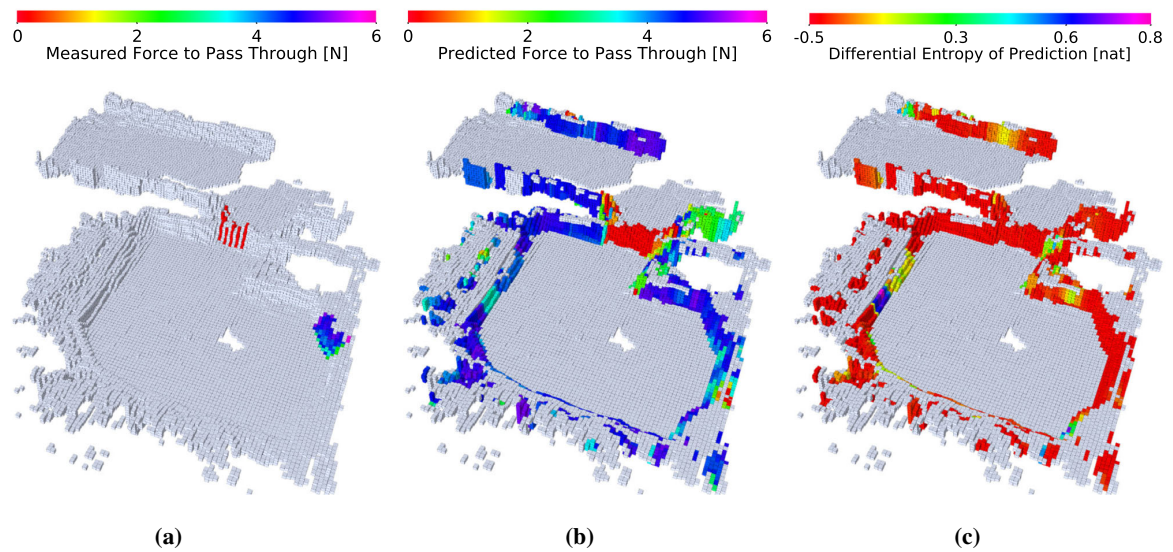

(a)   **(b)**   **(c)**

Fig. 4 **a** Example of haptic based traversability ground truth perceived by the robot as the measured force to pass through the obstacle. **b** The traversability prediction of the potential obstacles as the predicted force to pass through the obstacle and **c** the respective prediction differential entropy

cell ν. Since the haptic sensor utilizes a high sampling frequency, multiple different measurements are acquired for a single cell ν. Therefore, each cell uses a Kalman filter that fuses the measurements as

$$F_k(\nu) = \frac{\sigma_{\text{sensor}}^2 F_{k-1}(\nu) + \sigma_{k-1}^2(\nu) z_k^{\text{force}}(\nu)}{\sigma_{\text{sensor}}^2 + \sigma_{k-1}^2(\nu)},$$
$$\sigma_k^2(\nu) = \frac{\sigma_{\text{sensor}}^2 \sigma_{k-1}^2(\nu)}{\sigma_{\text{sensor}}^2 + \sigma_{k-1}^2(\nu)}, \qquad (9)$$

where $F_k(\nu)$ is the value reported by the *haptic* model for the cell ν after k measurements were assigned to the cell, $z_k^{\text{force}}(\nu)$ is the k-th measurement assigned to the cell, σ_{sensor}^2 is the bumper sensor measurement uncertainty, the initial force at the cell $F_0(\nu)$ equals the first measurement assigned to the cell, and the initial filter variance is $\sigma_0^2(\nu) = 1$. An example of the acquired traversability experience projected onto the elevation map (visualized in Fig. 3) is shown in Fig. 4a.

The haptic sensor is considered to provide the ground truth traversability measurements. Therefore, a binary value of the traversability p_{haptic} is utilized for cells where the ground truth measurements are available, with the traversability of the cell ν traversed by the haptic sensor being determined w.r.t. (2).

4.3 Traversability prediction

The ground-truth force measurements reported by the *haptic* module are limited to the particular obstacles the robot has interacted with. Hence, the ground truth traversability is relatively sparse. Therefore, the traversability *learning* module combines the exteroceptive and haptic information about the environment and determines the traversability for each observed cell ν where the ground truth is unavailable. The traversability predictor $\mathcal{P}$ is learned from the haptic experience that is extrapolated using the appearance A perceived by exteroceptive sensing.

The *learning* module learns to predict the force to pass through the observed potentially untraversable cell ν using its appearance. The terrain description $A(\nu)$ of each potentially untraversable cell ν is based on its color appearance. We advocate the use of simple descriptors, which are easy to compute and read. In scenarios with large, mono-color obstacles, we use a pair of the cell's a and b colors in the Lab color space computed from the RGB-colored elevation grid, as shown in Fig. 3a,

$$A^{\text{AB}}(\nu) = (a, b). \qquad (10)$$

The traversability predictor $\mathcal{P}$ is learned from each observed potentially untraversable cell ν that carriers the haptic force measurement $F(\nu)$ as

$$\mathcal{P} \leftarrow \text{learn}(\{(A(\nu), F(\nu))\}_{\{\nu\}}). \tag{11}$$

It predicts the force to pass through the potentially untraversable cell ν distributed as

$$\mathcal{N}(\mu_F(\nu, \mathcal{P}), \sigma_F^2(\nu, \mathcal{P})) \leftarrow \text{predict}(\nu, \mathcal{P}), \tag{12}$$

where the force prediction mean $\mu_F(\nu, \mathcal{P})$ determines the traversability prediction $p_{predict}(\nu, \mathcal{P})$ considering the force threshold F_{trav} as in (2). Besides, if the predictor $\mathcal{P}$ has not yet been learned, it assesses cells as untraversable. An example of the traversability prediction over the elevation grid map of the environment is shown in Fig. 4b.

The predicted distribution is utilized to estimate the expected information gain associated with sampling cells of unknown haptic traversability ground truth to steer the robotic exploration towards collecting the required information to improve the traversability model. The information expected from the haptic measurement at the cell ν with unknown haptic traversability $p_{haptic}(\nu)$ is approximated as the differential entropy of the respective predicted distribution (see Fig. 4c)

$$I_{predict}(\nu, \mathcal{P}) \approx H(\mathcal{N}(\mu_F, \sigma_F^2)) = \frac{1}{2} \log(2\pi e \sigma_F^2(\nu, \mathcal{P})). \tag{13}$$

Notice that for cell ν with the known haptic traversability ground truth $p_{haptic}(\nu)$, the ground truth with $\sigma_F^2 = 0$ is considered instead of the prediction, and thus no additional information can be gained, and the differential entropy at any such cell is undefined.

4.4 Gaussian process regressor

The traversability prediction is based on a GP regressor, briefly described here to make the paper self-contained. Given an observed function $f(x)$ with the noise ϵ

$$y = f(x) + \epsilon, \quad \epsilon \in \mathcal{N}(0, \sigma_n^2), \tag{14}$$

GP is a distribution over the functions (Rasmussen & Williams, 2006)

$$f(x) \sim \mathcal{GP}(m(x), K(x, x')) \tag{15}$$

where $m(x)$ and $K(x, x')$ are mean and covariance, respectively, defined as

$$m(x) = E[f(x)], \tag{16}$$
$$K(x, x') = E[(f(x) - m(x))(f(x') - m(x'))]. \tag{17}$$

Given the train data X and the test data X_*, the latent values f_* at X_* are

$$\mu(X_*) = K(X, X_*) \left[K(X, X) + \sigma_n^2 I \right]^{-1} y,$$
$$(\sigma(X_*))^2 = K(X_*, X_*)$$
$$- K(X, X_*)^T \left[K(X, X) + \sigma_n^2 I \right]^{-1} K(X, X_*), \tag{18}$$

where $K(X, X')$ is the covariance function. In this work, the used covariance function is the exponential kernel

$$K(\boldsymbol{x}, \boldsymbol{x}') = \sigma_{\exp}^2 \exp\left(-\frac{1}{l_{\exp}^2} \|\boldsymbol{x} - \boldsymbol{x}'\| \right). \tag{19}$$

In the considered exploration scenario, the model is learned online using only the available onboard computational resources of the robot. Hence, it is necessary to consider the computational requirements as the computation of GPs can generally be demanding. Therefore, the GP regressor is relearned from the accumulated traversability observations and terrain appearance with a fixed rate of 0.03 Hz. The real performance of the proposed terrain learning model is reported in Sect. 5 within the autonomous exploration that is briefly described in the following section.

4.5 Exploration scenario

The proposed terrain traversability approach with haptic and exteroceptive sensing is intended to model the robot's operational environment where some parts can look like obstacles in exteroceptive data but can be traversed. Since the robot environment is represented by the grid map $\mathcal{M}_{2.5D}$, the robot plans its paths through the environment w.r.t. the cost

$$c(\nu_a, \nu_b) = \|(\nu_a, \nu_b)\| + c_d(\mathcal{M}_{2.5D}), \tag{20}$$

where ν_a and ν_b are two cells that are 8-neighbors, the norm $\|(\nu_a, \nu_b)\|$ is the respective Euclidean distance between the cells' centers, and c_d is a non-negative cost. The cost c_d decreases with the distance from the closest untraversable cell to penalize robot presence close to the obstacles as in (Bayer & Faigl, 2019). The path cost c (20) is used to assess a cost of path ψ, and select the shortest path in the environment as

---

Algorithm 2: Learning the proposed traversability model in a robotic escape mission.

Input: $z^{\text{range}}_{1,\dots,m}$ – Colored range measurements; $z^{\text{force}}_{1,\dots,n}$ – Force measurements.
Output: $\mathcal{M}_{2.5D}$ – Colored elevation gridmap including traversability assesments for apparent obstacles; $\mathcal{P}$ – Traversability predictor for apparent obstacles.

```
1  M_{2.5D} ← ∅                                                          ▷ Initialize gridmap.
2  P ← ∅                                                                 ▷ Initialize predictor.
3  while exploration is running do
4      M_{2.5D} ← updateModel(M_{2.5D}, z^{range}_{i,...,i'})            ▷ Update the gridmap using the latest range measurements.
5      P ← updateModel(P, z^{force}_{j,...,j'}, M_{2.5D})               ▷ Update the predictor using the force measurements and terrain descriptors.
6      M_{2.5D} ← applyPredictions(P)                                    ▷ Apply traversability predictions to the gridmap.
7      for ν ∈ M_{2.5D} : p̂(ν) = 1 do                                   ▷ For each traversable cell ν (including predicted traversability).
8          compute I^{model}_{geom}(ν)                                   ▷ Compute the spatial information gained by observing from the cell w.r.t. (8).
9      if ∃ν ∈ M_{2.5D} : p̂(ν) = 1, I^{model}_{geom}(ν) > 0 then        ▷ Explore the spatial model if any information can be gained.
10         G_{geom} ← cluster(∀ν ∈ M_{2.5D} : I^{model}_{geom}(ν) > 0)  ▷ Create the spatial exploration goals.
11         ν* ← argmin_{ν_{geom}∈G_{geom}} c(ψ*(ν_{robot}, ν_{geom}))    ▷ Select the closest goal.
12     else if P is not learned then                                     ▷ Start exploring the predictor if it does not exist.
13         ν* ← argmin_{ν∈M_{2.5D}|p_{geom}(ν)=0} c(ψ*(ν_{robot}, ν))   ▷ Select the closest apparent obstacle as goal to start learning.
14     else                                                              ▷ Otherwise, continue exploring the predictor.
15         for ν ∈ M_{2.5D} : p_{geom}(ν) = 0 do                        ▷ For each potential obstacle ν.
16             compute I_{predict}(ν, P)                                 ▷ Compute the predictor information gained w.r.t. (13).
17         if max_{ν∈M_{2.5D}|p_{geom}(ν)=0} I_{predict}(ν, P) > I^{thr}_{predict} then  ▷ If enough information can be gained.
18             ν* ← argmax_{ν∈M_{2.5D}|p_{geom}(ν)=0} I_{predict}(ν, P)  ▷ Select the most informative cell as goal.
19         else                                                          
20             ν* ← ∅                                                    ▷ Otherwise, select no goal.
21     if ν* ≠ ∅ then                                                    
22         navigateTo(ν*)                                                ▷ Navigate to goal if it exists.
23     else                                                              
24         finishExploration()                                          ▷ Otherwise, end exploration.
25 return M_{2.5D}, P
```

$$\psi^*(\nu_{start}, \nu_{goal}) = \operatorname{argmin}_{\psi \in \Psi(\nu_{start}, \nu_{goal})} c(\psi),$$
$$c(\psi) = \sum_{i=1}^{|\psi|-1} c(\nu_i, \nu_{i+1}), \qquad (21)$$

where $\Psi(\nu_{start}, \nu_{goal})$ is the set of all possible paths from ν_{start} to ν_{goal}, and the path $\psi \in \Psi(\nu_{start}, \nu_{goal})$ is a path starting at ν_{start} and ending at ν_{end}. The cheapest path ψ^* is determined using the A* algorithm.

The exploration mission is considered as an escape-like scenario. The exploration procedure is overviewed in Algorithm 2, and it works as follows. The robot first explores the areas accessible without interacting with apparent obstacles. Only after all reachable areas are explored, the haptic sensor is actively used to learn the model of obstacle traversability. The exploration strategy selects the goal as

$$\nu^*_{explore} = \begin{cases} \nu^*_{geom} & \text{if } \nu^*_{geom} \text{ exists} \\ \nu^*_{predict} & \text{otherwise} \end{cases}, \qquad (22)$$

where the geometric exploration goal ν^*_{geom} is selected if it is possible to gain any additional information about the geometric model. The prediction model improvement goal $\nu^*_{predict}$ is selected otherwise. In particular, the robot selects the closest geometry exploration goal as

$$\nu^*_{geom} = \operatorname{argmin}_{\nu_{geom}\in\mathcal{G}_{geom}} c(\psi^*(\nu_{robot}, \nu_{geom})), \qquad (23)$$

where ν_{robot} is the cell corresponding to the current robot position. The goal to improve the prediction is selected either as the cell with the highest potential information gain about the prediction model, or the closest potentially untraversable cell if the prediction model is not yet learned

$$\nu^*_{predict} = \begin{cases} \nu^{*-I}_{predict} & \text{if } \mathcal{P} \text{ is learned} \\ \nu^{*-\Psi}_{predict} & \text{otherwise} \end{cases}, \qquad (24)$$

where the cell with the highest information gain potential is

$$\nu^{*-I}_{predict} = \operatorname{argmax}_{\nu\in\mathcal{M}_{2.5D}|p_{geom}(\nu)=0} I_{predict}(\nu, \mathcal{P}), \qquad (25)$$

and the closest potentially untraversable cell is

$$\nu^{*-\Psi}_{predict} = \operatorname{argmin}_{\nu\in\mathcal{M}_{2.5D}|p_{geom}(\nu)=0} c(\psi^*(\nu_{robot}, \nu)). \qquad (26)$$

Note that the area in the vicinity of the prediction goal is temporarily cleared as traversable in $\mathcal{M}_{2.5D}$ to allow the robot to approach the sampling location. Finally, the robot does not pursue prediction goals associated with less than $I^{\text{thr}}_{predict}$ information gained. Therefore, when there are no geometric
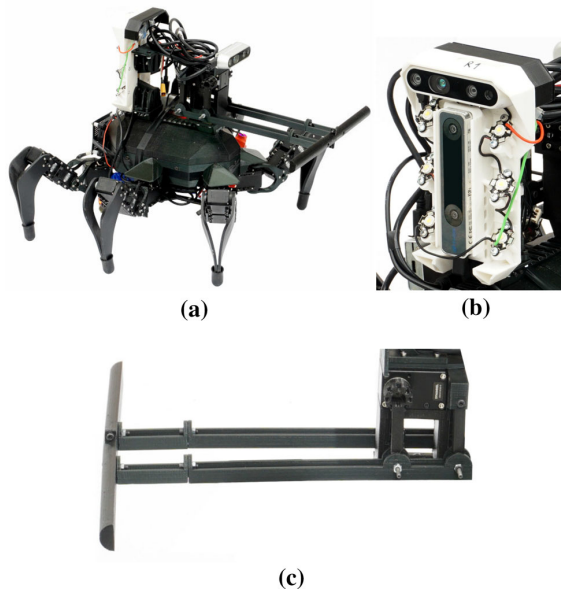
(a)

(b)

(c)

Fig. 5 **a** The hexapod walking robot used in the experimental deployment, **b** its rear-facing sensor rig, **c** and the hinge mechanism of its haptic bumper sensor

exploration goals and sampling no potentially untraversable cell is associated with more than $I_{predict}^{thr}$ information gained, the exploration stops.

5 Experimental results and discussion

The proposed system for active terrain traversability learning using visual and haptic cues has been experimentally verified in two scenarios. First, the robot is deployed in an escape-like exploration scenario to demonstrate active learning of passing through obstacles and thus explore areas that would remain inaccessible if only visual sensing would be used. The robot is deployed in an indoor office arena containing rigid obstacles the robot cannot traverse and non-rigid obstacles the robot can pass through. In the second scenario, we showcase the predictor in an outdoor setting with realistic vegetation.

Prior to the results from each scenario, a brief description of the utilized robot and its sensors are presented in Sect. 5.1. The two deployment scenarios are individually presented in Sects. 5.2 and 5.3, respectively. The results are further discussed in Sect. 5.4.

5.1 Robot and sensory equipment

The proposed system is deployed on the hexapod walking robot (Faigl & Čížek, 2019) shown in Fig. 5a. The

robot has six legs attached to its trunk, each comprising three Dynamixel AX12 servomotors. The robot, including its legs, can fit into a square with the side length of about 40 cm. The robot is equipped with exteroceptive sensor rigs to localize the robot and build the colored elevation map. The forward-facing rig comprises the Intel RealSense D435 RGB-D camera (D435 for short). The rear-facing camera rig holds another D435 and the Intel RealSense T265 tracking camera (T265 for short), see Fig. 5b. The localization using the rear-facing T265 is selected to avoid losing tracked features when the robot approaches obstacles.

The robot carries a haptic sensor designed as a bumper mounted on a parallelogram hinge, see Fig. 5c. The sensor is based on the Dynamixel XM430 servomotor that is actuated and set to return to a pre-set position. The servomotor provides torque measurements that are paired with the tabulated force values obtained by letting the bumper push on a force sensor prior to the deployment. The measured force values allow us to transform the bumper into a sensor measuring the force to pass through obstacles. The sensor has been individually calibrated for each presented experiment to avoid eventual bumper changes. The traversability threshold has been set to $F_{trav} = 2$ N.

Furthermore, the robot is equipped with two simple reflexes that help to sample the obstacle rigidity and traverse through non-rigid obstacles that can impede its visual sensors. First, when the robot gets close, within 0.25 m, to an obstacle, it tries to pass through by walking forward. The behavior is stopped either 10 s after leaving the obstacle vicinity or by triggering the second reflex when the bumper observes values higher than F_{trav}. In such a case, the robot samples the obstacle for 2 s to ensure that the observed value is not an outlier caused by bumper motion. Then, the robot engages a backward motion for 8 s to clear the obstacle.

The onboard computational resources are the Nvidia Jetson TX2 with 8 GB RAM running Robot Operating System (ROS) Melodic (Quigley et al., 2009) that demonstrated sufficient computational power according to the real computational requirements overviewed in Table 2. The proposed system is parametrized, as in Table 3. Similar to the exploration strategy proposed in Karolj et al. (2020), we use fixed values of the kernel hyper-parameters instead of optimizing them when recomputing the GP because the optimization process is computationally costly and thus not suitable for online deployments. We exploit that the ranges of the feature descriptors values and measured forces remain the same between the individual deployments and set the hyper-parameters empirically. In particular, the feature sensitivity can be adjusted via the kernel lengthscale with the intuition that it is possible to extrapolate roughly within the lengthscale distance of the known data. The kernel output variance is set so that the already sampled terrains report prediction entropies that are considered known w.r.t. $I_{predict}^{thr}$.

Table 2 System performance

Component	Update Rate [Hz]	CPU usage (%)
Exteroception (each D435)	1	25
Tactile sensing	300	4
Localization	200	7
Map building	2	7
Path planning	5	
Feature description	1	6
Learning and prediction	0.03	13
Locomotion	10	3

Table 3 System parametrization

Symbol	Description	Value	Reasoning
t_{trav}	Obstacle detection step height	0.12 m	Robot step height
δ_{sensor}	RGB-D sensor range	2 m	Properties of the sensor
d_v	Size of the squared grid cell of $\mathcal{M}_{2.5D}$	0.05 m	Size of the robot foothold
d_{cl}	Geometric goal cluster radius	1 m	Twice size of the robot
n_{cl}^{thr}	Geometric goal minimum cluster size	10	Set empirically
σ_{sensor}^2	Bumper sensor uncertainty in Kalman fuser	0.01	Bumper sensor calibration
σ_n^2	Gaussian Process noise variance σ_n^2	0.1	Set empirically
σ_{exp}	Gaussian Process exponential kernel σ_{exp}^2	1	Set empirically
l_{exp}	Gaussian Process exponential kernel (simple features) l_{exp}	1	Set empirically
	Gaussian Process exponential kernel (histogram features) l_{exp}	0.4	Set empirically
F_{trav}	Maximum force to push through obstacle	2 N	Properties of the robot
$I_{predict}^{thr}$	Miminum prediction model MI	0.26 nat	Set empirically ($\sigma_F^2 \approx 0.1$)
r_{hist}	Color histogram descriptor radius	0.15 m	Half of the bumper width

5.2 Exploration scenario

In the exploration scenario, the robot has been deployed in an arena split into three sections, see Fig. 6. The approximate size of the arena is 35 m^2. The robot cannot see the other sections of the arena from each section. The robot starts in a small arena section located in an office room with rigid brown wooden obstacles, rigid white walls, and purple fabric.

The fabric can be traversed to access a corridor containing the second and third arena sections. The two sections in the corridor contain similar obstacles as the first section. The corridor sections are divided by another purple fabric, enabling the robot which has already learned individual terrains' rigidity to identify the fabric as a traversable area. The goal of the experimental deployment is to explore all three sections and thus prove the ability of the proposed approach to sample and learn the force to pass through the individual obstacle types. The force sensor calibration in the exploration scenario is based on the values depicted in Table 4, which are used to compute a cubic spline to create a function of the force based on the torque measurements.
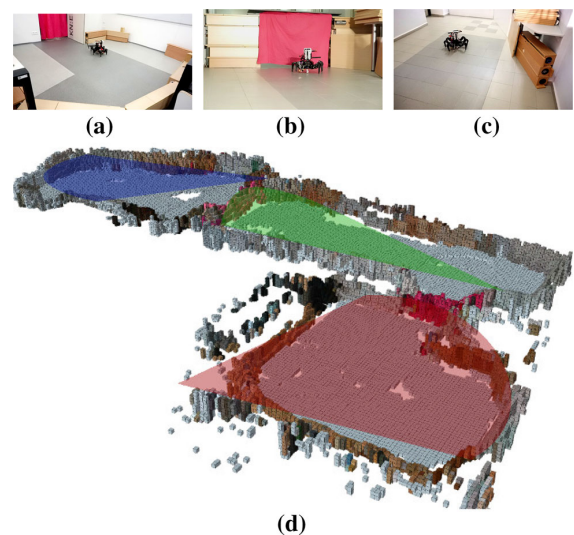

Fig. 6 The robot **a** in the first arena section, **b** leaving the second section, **c** exploring the third section; and **d** the map projection of the three respective image vision cones in the red, green and blue (Color figure online)

Table 4 Bumper sensor calibration values

Torque (Nm)	0	0.17	0.34	0.51	0.60	0.79	10*
Force (N)	0	1.05	2.02	3.10	4.09	6.34	10×10^4

*A limit value used in interpolation; not actually measured

Fig. 8 The **a** traversability and **b** color features in the arena during the exploration of the second section (Color figure online)

Particular maps showing the arena before the first sampling of the traversable fabric in the first section, during the exploration of the second section, and near the end of the experiment are shown in Figs. 7, 8, and 9, respectively. Note, a video of the therein depicted experimental run is also presented in Online Resource 1.

The robot behaved similarly in five deployment runs, of which we choose two particular runs to report the robot's behavior here. After exploring the initial section, the robot samples the obstacles. In general, if the robot has sampled the wall or a wooden obstacle first, it chooses to sample the purple curtain second, since its color in the Lab descriptor space is distant from the colors of the rigid obstacles and thus it has a high prediction model uncertainty. Hence, the robot walks through the curtain and enters the second section of the arena. There, the robot resumes spatial exploration and may attempt to traverse obstacles it considers traversable based on its previous experience. Since the robot has not necessarily sampled all the obstacles available in the first area, its traversability predictions might be too optimistic. It is expected behavior as it is a result of the incremental nature of the learning process. Nevertheless, the robot obtains new force ground truth when it attempts to traverse obstacles erroneously considered as traversable, thus further aiding the learning process as demonstrated in the alternative experimental run where the robot sampled the traversable fabric first, see Fig. 10.

The differences between the two experimental runs in the proportional representation of predictions considered as sure with regards to the uncertainty threshold $I_{predict}^{thr}$, and of the predictions thresholded as either traversable or untraversable

Fig. 9 The **a** traversability and **b** color features near the end of the experiment (Color figure online)
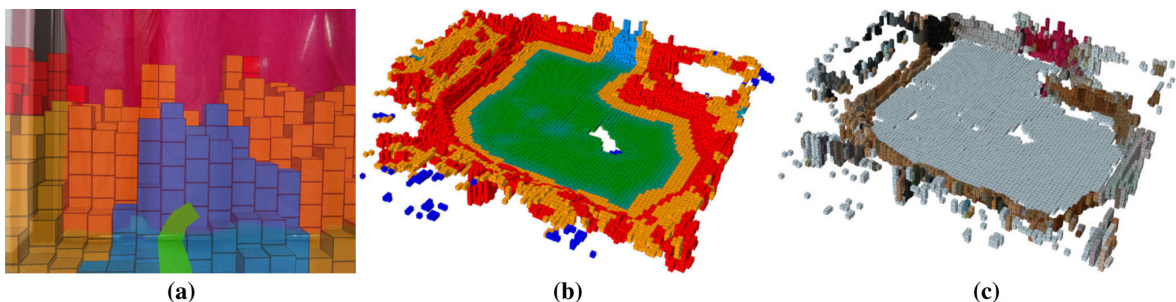

Fig. 7 The traversability **a** as seen from the robot and **b** its overview, and **c** color features in the arena before sampling the traversable fabric in the first section. N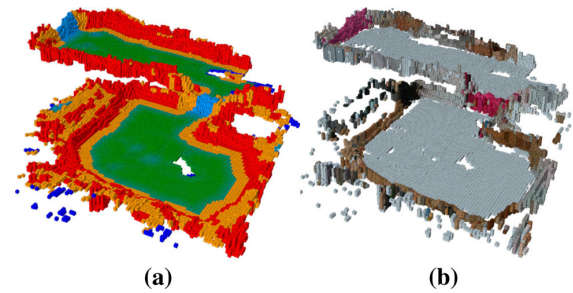ote, an obstacle is cleared as traversable even though the robot has not yet learned its rigidity because it needs to approach the obstacle to learn its traversability (Color figure online)

Fig. 10 The predicted traversability in the experimental run where the robot has sampled the traversable fabric first: **a** overly optimistic predictions after sampling the fabric and **b** corrected predictions after the robot has sampled other obstacles due to trying to traverse through them while exploring the corridor section
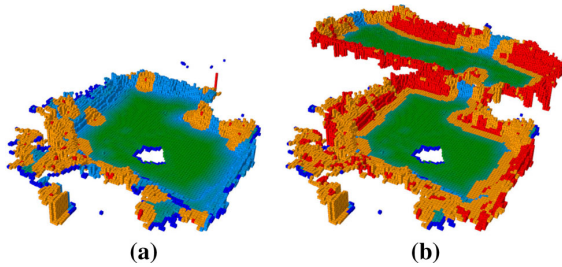
w.r.t. F_{trav}, can be seen in Fig. 11a, b, respectively. The evolution of the prediction entropy distribution is depicted in

Fig. 12. It can be seen that although the robot predicts a large portion of obstacles as traversable after only sampling the purple fabric in Fig. 11b, a large portion of these predictions is not sure. After the robot learns that some terrains are rigid by sampling them in the second area, sure untraversable predictions emerge, even though some uncertain traversable predictions remain, particularly in the first section of the map.

5.3 Outdoor scenario

In the outdoor scenario, the proposed system has been deployed in several locations in the Prokop Valley in Prague, Czech Republic. The bumper is used to collect the haptic data for several terrains in the area, which are paired with feature descriptors of the respective terrains. The bumper sensor calibration is depicted in Fig. 13.
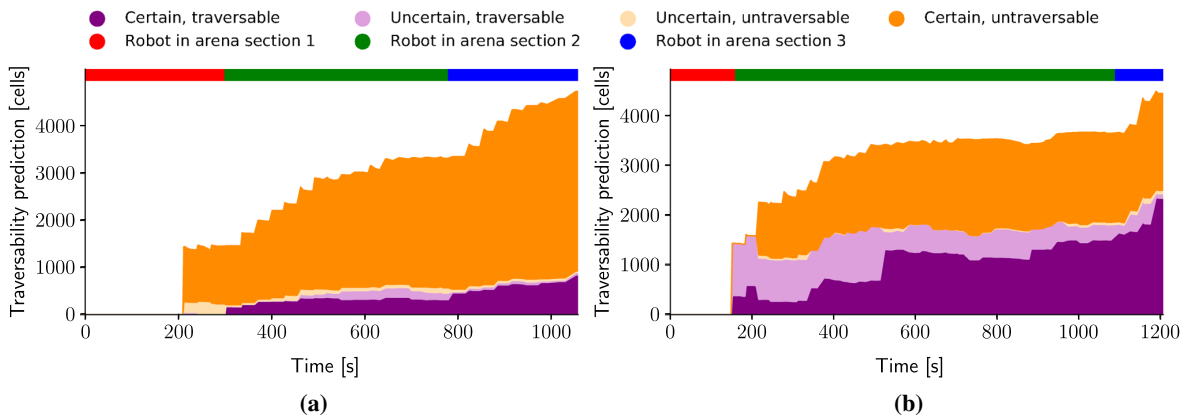

Fig. 11 Evolution of the predictions for potentially untraversable cells **a** in the main presented experimental run and **b** in the experimental run where the robot has sampled the traversable fabric first
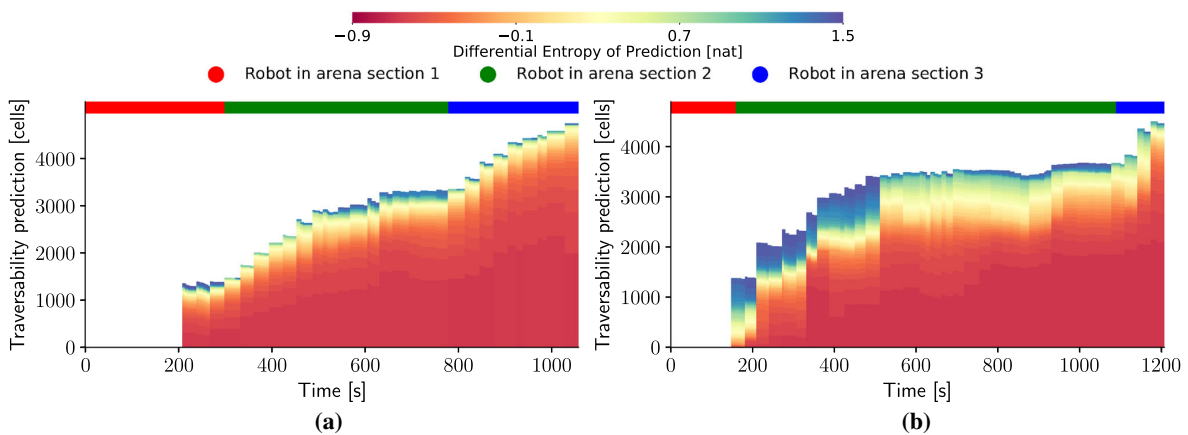

Fig. 12 Evolution of the predicted differential entropies for potentially untraversable cells in **a** the main presented experimental run; and in **b** the experimental run where the robot has sampled the traversable fabric first
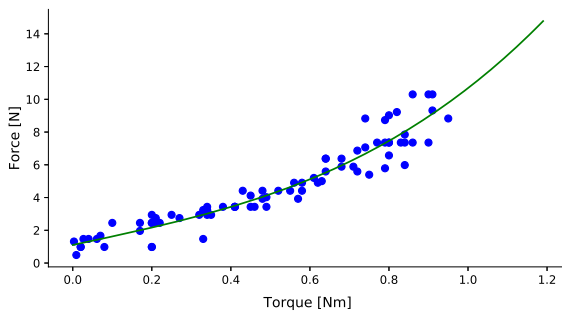
Fig. 13 Calibration data of the bumper sensor for the outdoor experiment. A cubic polynomial is fitted to the collected torque and force measurements

Fig. 15 The area (red circle) around a point of interest (red marker) signifying the cells used to compute the histogram terrain descriptor, projected on the elevation map with localized color measurements (Color figure online)
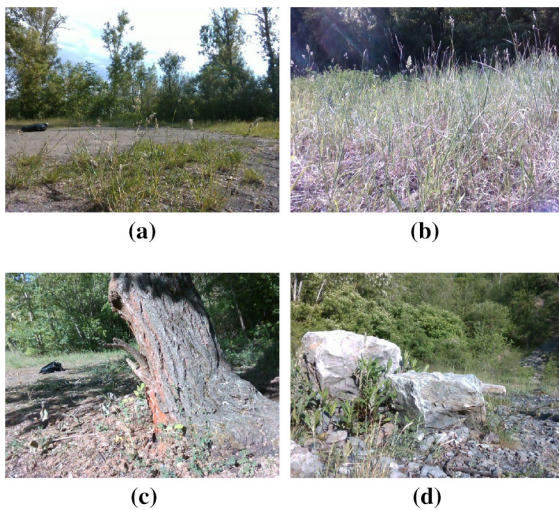

(a)   **(b)**

(c)   **(d)**

Fig. 14 The terrains used in the outdoor experiemnts: **a** sparse grass, **b** dense grass with hay, **c** tree trunk, **d** rocks

The terrains consist of several types of grass with varying density and appearance (some of them traversable), and a rigid tree trunk and rocks, see Fig. 14.

Since the terrains are more complex than in the indoor exploration, similarly to Belter et al. (2019), we utilize an alternative terrain descriptor in the outdoor scenario. The outdoor descriptors A^{hist} are based on color histograms, where each cell $v \in \mathcal{M}_{2.5D}$ is provided a 10-bit color by projecting the camera image to the $\mathcal{M}_{2.5D}$. Then, the color space is shrunk to 8 different colors: white, black, grey, blue, green, red, brown, and magenta. The relative amount of the cell color prototypes within the radius r_{hist} given by the half of the bumper size is used to build an 8-dimensional color histogram for each cell $v \in \mathcal{M}_{2.5D}$ as illustrated in Fig. 15. Besides, since the appearance descriptors differ from the indoor experiment in both scale and dimensionality, we adjust the kernel lengthscale, see Table 3.

The measurements from each terrain are split into testing and training sets. The algorithm is incrementally presented with training sets for the individual terrains, simulating a robot learning the terrains in a sequence. Table 5 shows the predictions on the respective testing sets in three alternative training sequences. The results suggest that after being presented with the particular terrain, the robot learns the terrain and reports entropy below the uncertainty threshold $I_{\text{predict}}^{\text{thr}}$; hence, marking the terrain as known.

5.4 Discussion

Based on the experimental deployment, we can conclude that the proposed robotic system actively learns traversable obstacles and can explore areas hidden behind such obstacles and thus explore a greater portion of such environment than a system considering all obstacles as untraversable. Unlike the state-of-the-art approaches that consider mobile robot interaction with non-rigid obstacles (Baleia et al., 2015; Kahn et al., 2021), the herein proposed system concerns mobile robot exploration, and the reasoning about the traversability of non-rigid obstacles is done in the context of the environment geometric map, which is built online as a part of the exploration process.

The difference is further manifested in the selected learning approach, where Kahn et al. (2021) uses an end-to-end network learned from a large amount of data gathered using a time-correlated random walk policy. Besides, visual-tactile sensing considered by Pearson et al. (2021) combines tactile whisker sensors with visual perception for place recognition in Multimodal Predictive Coding Network (MultiPredNet), a bio-inspired approach that comprises visual, tactile, and multi-sensory modules. While such approaches provide significant advantages in the form of navigation policy learning and crossmodal reconstruction, respectively,

Appendix B.4 – Prágr *et al.*: "Autonomous Exploration with Online Learning of Traversable yet Visually Rigid Obstacles" [C4], referenced on page 16.

Table 5 Outdoor scenario predictions: set test mean Prediction, set test mean prediction Entropy, set test RMSE

Sequence 1

	Sparse Grass			Dense Grass			Tree Trunk			Rock		
Ground Truth	1.106 ± 0.006			2.546 ± 0.377			13.869 ± 0.282			12.368 ± 0.480		
Learning Step	Prediction	Entropy	RMSE	Prediction	Entropy	RMSE	Prediction	Entropy	RMSE	Prediction	Entropy	RMSE
Sparse Grass	1.109	−**0.552**	0.007	1.111	0.902	1.484	1.107	1.418	12.765	1.109	1.398	11.270
Dense Grass	1.224	−**0.643**	0.270	2.643	−**0.534**	0.423	1.837	1.410	12.035	2.268	1.350	10.108
Tree Trunk	1.171	−**0.643**	0.196	2.711	−**0.535**	0.472	13.495	−**0.484**	0.749	7.394	1.334	4.999
Rock	1.206	−**0.647**	0.277	2.707	−**0.538**	0.428	13.509	−**0.488**	0.779	12.316	−**0.195**	0.642

Sequence 2

	Dense Grass			Rock			Sparse Grass			Tree Trunk		
Ground Truth	2.543 ± 0.365			12.430 ± 0.510			1.110 ± 0.018			13.356 ± 0.714		
Learning Step	Prediction	Entropy	RMSE	Prediction	Entropy	RMSE	Prediction	Entropy	RMSE	Prediction	Entropy	RMSE
Dense Grass	2.612	−**0.468**	0.477	2.478	1.329	9.967	2.112	1.207	1.006	2.441	1.411	10.938
Rock	2.641	−**0.471**	0.509	12.299	−**0.216**	0.671	3.937	1.201	2.983	7.499	1.389	5.893
Sparse Grass	2.639	−**0.517**	0.509	12.245	−**0.218**	0.783	1.114	−**0.653**	0.105	4.994	1.388	8.400
Tree Trunk	2.666	−**0.517**	0.542	12.402	−**0.226**	0.674	1.137	−**0.653**	0.228	13.566	−**0.626**	0.839

Sequence 3

	Tree Trunk			Rock			Sparse Grass			Dense Grass		
Ground Truth	13.789 ± 0.522			12.449 ± 0.534			1.106 ± 0.003			2.475 ± 0.360		
Learning Step	Prediction	Entropy	RMSE	Prediction	Entropy	RMSE	Prediction	Entropy	RMSE	Prediction	Entropy	RMSE
Tree Trunk	13.746	−**0.306**	0.742	14.026	1.393	1.641	14.013	1.413	12.909	14.238	1.393	11.769
Rock	13.658	−**0.325**	0.652	12.844	−**0.424**	0.933	12.676	1.364	11.574	11.866	1.202	9.414
Sparse Grass	13.400	−**0.328**	0.770	12.819	−**0.432**	0.889	1.265	−**0.538**	0.419	3.521	0.679	1.618
Dense Grass	13.394	−**0.333**	0.789	12.827	−**0.442**	0.869	1.245	−**0.548**	0.375	2.534	−**0.666**	0.399

The ground truth values are reported including the standard deviation. Each sequence represents a particular order in which the individual training sets (terrains) are passed to the predictor, and the respective predictions represent values after learning the particular terrain in the sequence. The split into training and testing sets is done randomly and differs for each sequence. The highligted values signify terrains considered known since their prediction entropy is below $t^{thr}_{predict}$

the herein proposed approach is focused on the problem of apparent-yet-non-rigid obstacles in mobile robot exploration. Hence, it learns only in areas associated with the high information gain concerning the prediction model and can learn online during the mission itself.

6 Conclusion

We present a system for online learning of the force needed to pass through obstacles employed in autonomous exploration to assess traversability in an environment with terrain that appears untraversable yet can be traversed. The robot actively learns the geometric model of its surroundings with model learning to predict the traversability of potentially obstructing terrains using a haptic sensor. Gaussian Process regressor is utilized for the force prediction representing the traversability of the potentially obstructing terrains. The robot actively navigates based on expected information gain from both the traversability predictor and the geometric model. The proposed system has been deployed in a fully autonomous experiment in an arena where the robot passed through an occluding non-rigid obstacle showing that the traversability properties have been successfully learned. Besides, we also show the performance of the predictor in an environment with real vegetation. The experimental results suggest that the robot successfully navigates an environment with non-rigid obstacles, chooses to explore areas that provide information for the rigidity and spatial models, and can discriminate different natural terrains.

In the future, we aim to develop a unified framework to combine traversal costs of visually traversable terrains and apparent obstacles, thus adding a class of exploration goals. Furthermore, we intend to generalize the proposed robot-terrain interaction modeling to encompass traversable obstacles, rigid appearing terrains that do not support the robot, and terrains that change appearance after physical interaction. The exploration could also be focused on constraints such as the robot's battery capacity. Such extensions lead to deploying the proposed approach in non-myopic scenarios, where the robot considers its plans further than the immediate next navigation goal (waypoint). Hence, we aim to exploit multi-goal path planning (Faigl & Kulich, 2013) for such extended exploration scenarios.

Declarations

Conflict of interest The authors declare that they have no relevant financial or non-financial interests to disclose.
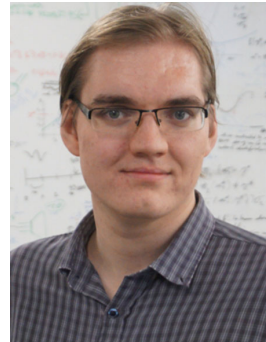
References

Baleia, J., Santana, P., & Barata, J. (2015). On exploiting haptic cues for self-supervised learning of depth-based robot navigation affordances. *Journal of Intelligent & Robotic Systems, 80*(3–4), 455–474. https://doi.org/10.1007/s10846-015-0184-4.

Bayer, J., & Faigl, J. (2019). On autonomous spatial exploration with small hexapod walking robot using tracking camera intel realsense T265. In *European conference on mobile robots (ECMR)* (pp. 1–6). https://doi.org/10.1109/ECMR.2019.8870968

Bayer, J., & Faigl, J. (2020). Speeded up elevation map for exploration of large-scale subterranean environments. In *2019 modelling and simulation for autonomous systems (MESAS)* (pp. 190–202). https://doi.org/10.1007/978-3-030-43890-6_15

Belter, D., Wietrzykowski, J., & Skrzypczyński, P. (2019). Employing natural terrain semantics in motion planning for a multi-legged robot. *Journal of Intelligent & Robotic Systems, 93*(3), 723–743. https://doi.org/10.1007/s10846-018-0865-x.

Biber, P., & Duckett, T. (2005). Dynamic maps for long-term operation of mobile service robots. In *Robotics: Science and systems (RSS)*. https://doi.org/10.15607/RSS.2005.I.003

Bourgault, F., Makarenko, A. A., Williams, S. B., Grocholsky, B., & Durrant-Whyte, H. F. (2002). Information based adaptive robotic exploration. In *IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 540–545). https://doi.org/10.1109/IRDS.2002.1041446

Bradley, D. M., Chang, J. K., Silver, D., Powers, M., Herman, H., Rander, P., & Stentz, A. (2015). Scene understanding for a high-mobility walking robot. In *IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 1144–1151). https://doi.org/10.1109/IROS.2015.7353514

Brown, D., & Webster, G. (2010). Now a stationary research platform, nasa's mars rover spirit starts a new chapter in red planet scientific studies. NASA Press Release

Brunner, M., Brüggemann, B., & Schulz, D. (2013). Rough terrain motion planning for actuated, tracked robots. In *International conference on agents and artificial intelligence (ICAART)* (pp. 40–61). Springer. https://doi.org/10.1007/978-3-662-44440-5_3

Burgard, W., Cremers, A. B., Fox, D., Hähnel, D., Lakemeyer, G., Schulz, D., et al. (1999). Experiences with an interactive museum tour-guide robot. *Artificial Intelligence, 114*(1), 3–55. https://doi.org/10.1016/S0004-3702(99)00070-3.

Carrillo, H., Dames, P., Kumar, V., & Castellanos, J. A. (2018). Autonomous robotic exploration using a utility function based on Rényi's general theory of entropy. *Autonomous Robots, 42*(2), 235–256. https://doi.org/10.1007/s10514-017-9662-9.

Cunningham, C., Nesnas, I. A., & Whittaker, W. L. (2019). Improving slip prediction on Mars using thermal inertia measurements. *Autonomous Robots, 43*(2), 503–521. https://doi.org/10.1007/s10514-018-9796-4.

Faigl, J., & Čížek, P. (2019). Adaptive locomotion control of hexapod walking robot for traversing rough terrains with position feedback only. *Robotics and Autonomous Systems, 116,* 136–147. https://doi.org/10.1016/j.robot.2019.03.008.

Faigl, J., & Kulich, M. (2013). On determination of goal candidates in frontier-based multi-robot exploration. In *European conference on mobile robots (ECMR)* (pp. 210–215). https://doi.org/10.1109/ECMR.2013.6698844

Faigl, J., & Kulich, M. (2015). On benchmarking of frontier-based multi-robot exploration strategies. In *European conference on mobile robots (ECMR)* (pp. 1–8). https://doi.org/10.1109/ECMR.2015.7324183

Giguere, P., & Dudek, G. (2008). Clustering sensor data for terrain identification using a windowless algorithm. In *Robotics: Science and systems (RSS)*. https://doi.org/10.15607/RSS.2008.IV.004

Gu, J., Cao, Q., & Huang, Y. (2008). Rapid traversability assesment in 2.5 d grid based map on rough terrain. *International Journal of Advanced Robotic Systems*. https://doi.org/10.5772/6233.

Halodová, L., Dvořáková, E., Majer, F., Vintr, T., Mozos, O. M., Dayoub, F., & Krajník, T. (2019). Predictive and adaptive maps for long-term visual navigation in changing environments. In *IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 7033–7039). https://doi.org/10.1109/IROS40897.2019.8967994

Homberger, T., Bjelonic, M., Kottege, N., & Borges, P. V. K. (2016). Terrain-dependant control of hexapod robots using vision. In *International symposium on experimental robotics (ISER)* (pp. 92–102). Springer. https://doi.org/10.1007/978-3-319-50115-4_9

Homberger, T., Wellhausen, L., Fankhauser, P., & Hutter, M. (2019). Support surface estimation for legged robots. In *IEEE international conference on robotics and automation (ICRA)* (pp. 8470–8476). https://doi.org/10.1109/ICRA.2019.8793646

Jadidi, M. G., Miro, J. V., & Dissanayake, G. (2018). Gaussian processes autonomous mapping and exploration for range-sensing mobile robots. *Autonomous Robots, 42*(2), 273–290. https://doi.org/10.1007/s10514-017-9668-3.

Kahn, G., Abbeel, P., & Levine, S. (2021). BADGR: An autonomous self-supervised learning-based navigation system. *Robotics and Automation Letters, 6*(2), 1312–1319. https://doi.org/10.1109/LRA.2021.3057023.

Karolj, V., Viseras, A., Merino, L., & Shutin, D. (2020). An integrated strategy for autonomous exploration of spatial processes in unknown environments. *Sensors, 20*(13), 3663.

Kottege, N., Parkinson, C., Moghadam, P., Elfes, A., & Singh, S. P. N. (2015). Energetics-informed hexapod gait transitions across terrains. In *IEEE international conference on robotics and automation (ICRA)* (pp. 5140–5147). https://doi.org/10.1109/ICRA.2015.7139915

Kragh, M., Jørgensen, R. N., & Pedersen, H. (2015). Object detection and terrain classification in agricultural fields using 3D lidar data. In *International conference on computer vision systems (ICVS)* (pp. 188–197). https://doi.org/10.1007/978-3-319-20904-3_18

Krüsi, P., Bosse, M., & Siegwart, R. (2016). Driving on point clouds: Motion planning, trajectory optimization, and terrain assessment in generic nonplanar environments. *Journal of Field Robotics, 34*(5), 940–984. https://doi.org/10.1002/rob.21700.

Lalonde, J. F., Vandapel, N., Huber, D. F., & Hebert, M. (2006). Natural terrain classification using three-dimensional ladar data for ground robot mobility. *Journal of Field Robotics, 23*(10), 839–861. https://doi.org/10.1002/rob.20134.

Luo, W., & Sycara, K. (2018). Adaptive sampling and online learning in multi-robot sensor coverage with mixture of Gaussian processes. In *IEEE international conference on robotics and automation (ICRA)* (pp. 6359–6364). https://doi.org/10.1109/ICRA.2018.8460473

Makarenko, A. A., Williams, S. B., Bourgault, F., & Durrant-Whyte, H. F. (2002). An experiment in integrated exploration. In *IEEE/RSJ international conference on intelligent robots and systems (IROS)* (vol. 1, pp. 534–539). https://doi.org/10.1109/IRDS.2002.1041445

McGhee, R. B., & Frank, A. A. (1968). On the stability properties of quadruped creeping gaits. *Mathematical Biosciences, 3,* 331–351. https://doi.org/10.1016/0025-5564(68)90090-4.

Moravec, H., & Elfes, A. (1985). High resolution maps from wide angle sonar. *IEEE International Conference on Robotics and Automation (ICRA), 2,* 116–121. https://doi.org/10.1109/ROBOT.1985.1087316.

O'Meadhra, C., Tabib, W., & Michael, N. (2019). Variable resolution occupancy mapping using gaussian mixture models. *Robotics and Automation Letters, 4*(2), 2015–2022. https://doi.org/10.1109/LRA.2018.2889348.

Otsu, K., Ono, M., Fuchs, T. J., Baldwin, I., & Kubota, T. (2016). Autonomous terrain classification with co- and self-training approach. *Robotics and Automation Letters, 1*(2), 1–6. https://doi.org/10.1109/LRA.2016.2525040.

Papadakis, P. (2013). Terrain traversability analysis methods for unmanned ground vehicles: A survey. *Engineering Applications of Artificial Intelligence, 26*(4), 1373–1385. https://doi.org/10.1016/j.engappai.2013.01.006.

Pearson, M. J., Dora, S., Struckmeier, O., Knowles, T. C., Mitchinson, B., Tiwari, K., et al. (2021). Multimodal representation learning for place recognition using deep Hebbian predictive coding. *Frontiers in Robotics and AI*. https://doi.org/10.3389/frobt.2021.732023.

Petrou, Z., Manakos, I., Stathaki, T., Mücher, C., & Adamo, M. (2015). Discrimination of vegetation height categories with passive satellite sensor imagery using texture analysis. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 8*(4), 1442–1455. https://doi.org/10.1109/JSTARS.2015.2409131.

Prágr, M., Čížek, P., & Faigl, J. (2018a). Incremental learning of traversability cost for aerial reconnaissance support to ground units. In *Modelling and simulation for autonomous systems (MESAS)*. Springer. https://doi.org/10.1007/978-3-030-14984-0_30

Prágr, M., Čížek, P., & Faigl, J. (2018b). Cost of transport estimation for legged robot based on terrain features inference from aerial scan. In *IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 1745–1750). https://doi.org/10.1109/IROS.2018.8593374

Prágr, M., Čížek, P., Bayer, J., & Faigl, J. (2019). Online incremental learning of the terrain traversal cost in autonomous exploration. In *Robotics: Science and systems (RSS)*. https://doi.org/10.15607/RSS.2019.XV.040

Prágr, M., & Faigl, J. (2019). Benchmarking incremental regressors in traversal cost assessment. In *International conference on artificial neural networks (ICANN)* (pp. 685–697). https://doi.org/10.1007/978-3-030-30487-4_52

Prágr, M., Váňa, P., & Faigl, J. (2020). Aerial reconnaissance and ground robot terrain learning in traversal cost assessment. In *2019 modelling and simulation for autonomous systems (MESAS)* (pp. 3–10). https://doi.org/10.1007/978-3-030-43890-6_1

Quigley, M., Conley, K., Gerkey, B. P., Faust, J., Foote, T., Leibs, J., Wheeler, R., & Ng, A. Y. (2009). ROS: An open-source robot operating system. In *ICRA workshop on open source software* (pp. 1–6)

Ramos, F., & Ott, L. (2016). Hilbert maps: Scalable continuous occupancy mapping with stochastic gradient descent. *International Journal of Robotics Research, 35*(14), 1717–1730. https://doi.org/10.1177/0278364916684382.

Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning: Adaptive computation and machine learning*. MIT Press.

Rényi, A. (1961). On measures of entropy and information. In *Berkeley symposium on mathematical statistics and probability* (vol. 1, pp. 547–561). https://projecteuclid.org/euclid.bsmsp/1200512181

Rothrock, B., Kennedy, R., Cunningham, C., Papon, J., Heverly, M., & Ono, M. (2016). SPOC: Deep learning-based terrain classification for mars rover missions. In *AIAA SPACE 2016, American Institute of aeronautics and astronautics*. https://doi.org/10.2514/6.2016-5539

Ruiz, A. V., & Olariu, C. (2015). A general algorithm for exploration with Gaussian processes in complex, unknown environments. In

IEEE international conference on robotics and automation (ICRA) (pp. 3388–3393). https://doi.org/10.1109/ICRA.2015.7139667

Singh, A., Krause, A., Guestrin, C., Kaiser, W., & Batalin, M. (2007). Efficient planning of informative paths for multiple robots. In *International joint conference on artifical intelligence* (pp. 2204–2211). https://doi.org/10.5555/1625275.1625631

Sofman, B., Lin, E., Bagnell, J. A., Cole, J., Vandapel, N., & Stentz, A. (2006). Improving robot navigation through self-supervised online learning. *Journal of Field Robotics, 23*(11–12), 1059–1075. https://doi.org/10.1002/rob.20169.

Srinivas, N., Krause, A., Kakade, S., & Seeger, M. (2010). Gaussian process optimization in the bandit setting: no regret and experimental design. In *Intl. Conf. international conference on machine learning (ICML)*. Haifa, Israel (pp. 1015–1022). https://icml.cc/conferences/2010/papers/422.pdf

Stachniss, C., Grisetti, G., & Burgard, W. (2005). Information gain-based exploration using rao-blackwellized particle filters. In *Robotics: Science and systems (RSS), Robotics: Science and systems foundation.* https://doi.org/10.15607/RSS.2005.I.009

Stelzer, A., Hirschmüller, H., & Görner, M. (2012). Stereo-vision-based navigation of a six-legged walking robot in unknown rough terrain. *International Journal of Robotics Research, 31*(4), 381–402. https://doi.org/10.1177/0278364911435161.

Tabib, W., Goel, K., Yao, J., Dabhi, M., Boirum, C., & Michael, N. (2019). Real-time information-theoretic exploration with Gaussian mixture model maps. In *Robotics: Science and systems (RSS)* (vol. 15). https://doi.org/10.15607/RSS.2019.XV.061

Tiwari, K., Xiao, X., Malik, A., & Chong, N. Y. (2019). A unified framework for operational range estimation of mobile robots operating on a single discharge to avoid complete immobilization. *Mechatronics, 57,* 173–187. https://doi.org/10.1016/j.mechatronics.2018.12.006.

Tucker, V. A. (1975). The energetic cost of moving about: Walking and running are extremely inefficient forms of locomotion: Much greater efficiency is achieved by birds, fish-and bicyclists. *American Scientist, 63*(4), 413–419.

Ünsalan, C., & Boyer, K. L. (2004). Linearized vegetation indices based on a formal statistical framework. *Transactions on Geoscience and Remote Sensing, 42*(7), 1575–1585. https://doi.org/10.1109/TGRS.2004.826787.

Vasudevan, S., Ramos, F., Nettleton, E., Durrant-Whyte, H., & Blair, A. (2009). Gaussian Process modeling of large scale terrain. In *IEEE international conference on robotics and automation (ICRA)* (pp. 1047–1053). IEEE. https://doi.org/10.1002/rob.20309

Yamauchi, B. (1997). A frontier-based approach for autonomous exploration. In *CIRA* (pp. 146–151). IEEE. https://doi.org/10.1109/CIRA.1997.613851

Miloš Prágr is a Ph.D. student at the Faculty of Electrical Engineering (FEE), Czech Technical University in Prague (CTU), Czech Republic. In 2018, he received his master's (Ing.) degree in computer vision at FEE, CTU. Since then, he has worked under the supervision of Jan Faigl, who heads the Computational Robotics Laboratory, Artificial Intelligence Center, FEE, CTU. His research is focused on modeling traversability of mobile robots over terrains, with a particular interest in traversability models that are learned online during the robot's deployment.

Jan Bayer is a Ph.D. student at the Czech Technical University in Prague (CTU), Czech Republic. In 2019, he received his master's (Ing.) degree in cybernetics and robotics at the Faculty of Electrical Engineering (FEE), CTU. His master thesis won the first place in the IT Spy competition, which is awarded by the Czech and Slovak ACM chapters. Since then, he has worked under the supervision of Jan Faigl, who heads the Computational Robotics Laboratory, Artificial Intelligence Center, FEE, CTU. He has participated in DARPA SubT Challenge as a member of the CTU-CRAS-NORLAB team. His research is focused on mobile robot exploration, with a particular interest in multi-robot scenarios.

Jan Faigl is a professor of computer science at the Faculty of Electrical Engineering (FEE), Czech Technical University in Prague (CTU). He received his Ph.D. (2010) in Artificial Intelligence and Biocybernetics and Ing. (2003) in Cybernetics from CTU. In 2013/2014, he was a Fulbright visit scholar at the University of Southern California. He had been awarded the Antonin Svoboda Award from the Czech Society for Cybernetics and Informatics in 2011. He served as the guest editor of the special issue on "Online decision making in Multi-Robot Coordination" in the Autonomous Robots journal. He currently serves as the associate editor of the IEEE Transactions on Automation Science and Engineering (T-ASE). Since 2013, he is leading the Computational Robotics Laboratory (http://comrob.fel.cvut.cz) within the Artificial Intelligence Center (http://aic.fel.cvut.cz). He is co-founder of the Center for Robotics and Autonomous Systems (CRAS) - http://robotics.fel.cvut.cz. He participated in DARPA SubT Challenge as co-leader of the CTU-CRAS-NORLAB team. In 2021, he received Amazon Research Award. His current research interests include combinatorial motion planning, robotic information gathering, long-term autonomous missions with incremental online learning, autonomous navigation, aerial systems, and multi-legged locomotion control.