Robotics and Autonomous Systems 168 (2023) 104494

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Traversability analysis for off-road environments using locomotion experiments and earth observation data

Matthias Eder [a,*], Raphael Prinz [b], Florian Schöggl [c], Gerald Steinbauer-Wagner [a]

[a] Graz University of Technology, Institute of Software Technology, Inffeldgasse 16b/II, Graz, 8010, Styria, Austria
[b] Graz University of Technology, Institute of Geodesy, Steyrergasse 30, Graz, 8010, Styria, Austria
[c] Pentamap GmbH, Petersbachstraße 28, Graz, 8042, Styria, Austria

ARTICLE INFO

Article history:
Received 17 January 2023
Received in revised form 28 April 2023
Accepted 17 July 2023
Available online 20 July 2023

Keywords:
Traversability analysis
Locomotion
Mobile robots
Cost map
Off-road navigation
Earth observation
Data-driven

ABSTRACT

In recent years, the navigation capabilities of mobile robots in off-road environments have increased significantly, opening up new potential applications in a variety of settings. By accurately identifying different types of terrain in unstructured environments, safe automated navigation can be supported. However, to enable safe path planning and execution, the traversability costs of the terrain types need to be accurately estimated. Such estimations are often performed manually by experts who possess information about the environment and are familiar with the capabilities of the robotic system or using simplified experiments. In this paper, we present an automated pipeline for generating traversability costs that use recorded locomotion data from a realistic experiment and descriptive information on the terrain obtained from earth observation data. The main contribution is that the cost estimation for different terrain types is based on locomotion data obtained in realistic standardized experiments. Moreover, by repeating the experiments with different robot systems we are easily able to reflect the actual capabilities of the systems. Experiments were conducted in an alpine off-road environment to record locomotion data of four different robot systems and to investigate the performance and validity of the proposed pipeline. The recorded locomotion data for the different robots are publicly available at https://robonav.ist.tugraz.at/data/

1. Introduction

Navigation for mobile robots is a highly active research field where several disciplines meet, ranging from path and motion planning over robot localization and perception to environment mapping. While these problems are tackled for robotic systems in various settings, the representation and perception of the robot's environment are particularly important for off-road navigation [1,2]. Therefore, it is necessary to know the terrain in which the robot operates and to analyze its traversability [3]. For terrain classification, information about the environment is collected and segmented into different terrain types. Nowadays, this is typically done using Deep Learning methods on visual [4,5] or LiDAR data [6]. After environment segmentation and classification, the traversability of the terrain classes needs to be estimated to allow the planning of a safe route for the robot. To take into account the traversability of the terrain during navigation, this information is typically represented as a cost map [7]. The costs for traversing terrain classes are often estimated manually by a human expert and usually do not fully reflect the capabilities of a specific robot system and its performance on a particular terrain [8]. Although methods to help estimate the traversability costs exist (e.g. [9]), they depend on parameters that need to be tuned manually by a human expert. While human experts are quite capable of judging the general traversability, it is much harder to quantify traversal costs for different terrain classes. For example, it is difficult to quantify traversability for gravel roads versus mowed grass. This estimation becomes even more difficult when multidimensional terrain information such as different slope angles or weather information (e.g., dry, wet, snow) is provided by the terrain classification. Considering such information during the navigation can be very helpful as these parameters heavily affect locomotion performance. In addition, it is hard for human experts to accurately assess the capabilities of a robotic system, as tracked, wheeled or legged robots show quite different performances on various terrains. Controlled experiments in a laboratory that evaluate such capabilities are a first step in estimating traversability, but hardly represent the actual capabilities of a robot in very diverse off-road environments [10]. Approaches that make use of Deep Learning methods typically perform better, but often use highly simplified experiments that do not fully reflect the actual traversability of a robotic system [11,12].

* Corresponding author.
E-mail addresses: matthias.eder@ist.tugraz.at (M. Eder), raphael.prinz@tugraz.at (R. Prinz), florian.schoeggl@pentamap.com (F. Schöggl), steinbauer@ist.tugraz.at (G. Steinbauer-Wagner).

https://doi.org/10.1016/j.robot.2023.104494

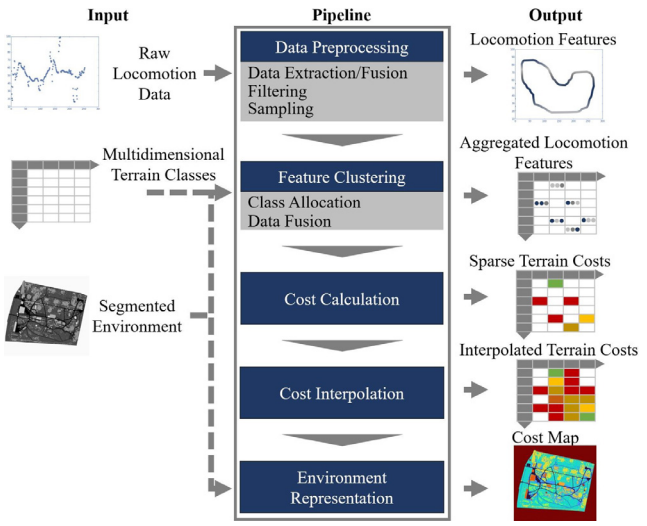

Fig. 1. Overview of the proposed processing pipeline. Locomotion data is processed and clustered based on terrain classes to generate locomotion features. These features are used to estimate costs for each terrain class (green = low cost, red = high cost) which are then interpolated to generate a complete cost map. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

This work aims to overcome these issues and proposes an extension of a pipeline for traversability analysis and cost estimation for a robotic system on multidimensional terrain classes that utilizes recorded locomotion data from a realistic environment, as previously presented in [13], by adding a more detailed description of the algorithm, new insights, and new results. Moreover, we define standardized locomotion experiments to obtain informative locomotion data for different robot systems in these environments. The pipeline uses locomotion data, multidimensional terrain classes, and environment segmentation to compute robot-dependent traversability costs. In multidimensional terrain classes, each dimension represents a property of the terrain (e.g. slope information, weather information, or land cover type). Locomotion data contains raw information about the performance and quality of a particular motion of a robot. For example, this information may represent general inertial and position data or may represent hardware-specific measurements such as power consumption or motor temperatures. A requirement for the pipeline to work is that the individual motion information could be linked to the corresponding terrain classes, that are represented in the segmented environment. For this, a localization approach (e.g. RTK-GNSS) that delivers sufficient precise results at a sufficient rate in the environment's frame is needed. Another requirement is that the recording experiments are conducted in a way that for (all) instances of the multidimensional terrain classes sufficient locomotion data was sampled. Using the recorded locomotion data, locomotion features are extracted (e.g. smoothness or velocity of a motion), clustered by their corresponding terrain class, and condensed to a set of aggregated locomotion features which represent the traversability properties of a given terrain class. In the next step, the aggregated locomotion features are used within a linear weighted cost function, to compute the traversability of the terrain. Fig. 1 gives an overview of the proposed pipeline. The resulting costs for a terrain class are represented in Fig. 1 as a colored grid which indicates how well a given terrain class is traversable (green = low cost/easy to traverse, red = high cost/not traversable). Since it cannot be expected that every multidimensional terrain class can be sampled sufficiently when recording the locomotion dataset, the calculated terrain costs are interpolated for unseen classes if some terrain dimensions show an order relation (e.g. for slope classes). The main contributions of this paper are (1) a pipeline to estimate the traversability costs for a specific robot in an off-road environment holding multidimensional terrain classes, (2) interpolating the optimized costs to uncovered terrain classes that have an order relation to computed terrain costs, and (3) the description and provision of realistic off-road locomotion datasets from four robots with different locomotion capabilities.

The remainder of this paper is structured as follows. Section 2 discusses related research on the topics of terrain classification and traversability analysis for robot navigation. In Section 3, the proposed method is described in detail. Experiments to validate the pipeline are presented in Section 4 and discussed in Section 5. Section 6 concludes the paper.

2. Related work

This section discusses related work in the field of traversability analysis and terrain classification. While both terms are sometimes used interchangeably (e.g. [14,15]), we see two different concepts. We see terrain classification as a more general term and define it as the provision of semantic information about the environment to improve the robot's understanding of its environment [16]. Traversability analysis, on the other hand, aims to provide a measure of how well a robot is able to navigate on certain terrain [17] and can be seen as a user of terrain classification. To further distinguish the two terms, this section discusses related research in the field of terrain classification and traversability analysis.

2.1. Terrain classification

To provide semantic information about a robot's environment, this information needs to be extracted from data about the environment. This is usually done by using on-board sensors of the robot such as camera [16,18] or LiDAR [6] but can also be conducted using external environment information, e.g. derived from satellite imagery [19,20]. Terrain classification overlaps strongly with the problem of semantic segmentation. Thus, many approaches use Machine Learning such as Convolutional Neural Networks (CNN) trained on annotated camera images [4,5]. Annotations for images are usually generated by manually annotating image by image (e.g. in [21]) which is time-consuming and error-prone. To support the manual process, different tools are available, such as edge detection to track the annotations over the following video frames [22] or post-processing techniques to densify gained annotations retrieved by the annotation tools [23]. As such methods heavily depend on a large amount of training data and to enable the comparability of classification approaches, open-source datasets are often provided [21,24,25]. Alternatives to camera images for training are annotated 3D point clouds derived from LiDAR sensors [6,26] or even inertial measurements [27].

2.2. Traversability analysis for robot navigation

According to [28], the analysis of the traversability can be based on three types: (1) proprioceptive-based (e.g. [10,17]), (2) geometry-based (e.g. [29]), and (3) appearance-based (e.g. [18]). While the latter two types typically use camera and LiDAR information to estimate the traversability of the robot, the first type uses only intrinsic information of a robot (e.g. Inertial measurements) to represent a given environment. In this work, we focus on the traversability analysis using proprioceptive data, namely locomotion information, which is combined with earth

observation data to estimate the traversability of a given terrain class.

Traversability analysis can be conducted with the goal of either providing binary (traversable/non-traversable) [16], continuous [3], or discrete cost estimates [9] following a given objective. There are multiple objectives for estimating the traversability of a terrain such as slope, speed, smoothness, or energy consumption [30]. While some work focuses on the estimation of a single objective (e.g. [12]), others define the traversability as a combination of multiple objectives (e.g. [29]). However, most of the approaches have in common that they are designed only for a given set of objectives and can hardly be applied to other objectives. Our proposed pipeline can estimate the traversability of a terrain given any number of objectives and provides discrete cost estimates for a given terrain.

The estimation of the traversability of a (segmented) environment is usually done either manually [8], with the help of statistical analysis [9], or using Deep Learning approaches [3,11,17]. While all these approaches show acceptable performance in estimating the traversability of a given terrain, the capabilities of the used robot are neglected, as robot-specific parameters are assumed to be provided. Estimating the traversability costs purely on robot-specific motion characterization was done in [31,32]. In our work, we take the capabilities of a robot into account and use environmental information to extract as well as compare traversability estimates for four different robot systems.

Methods for traversability analysis heavily rely on data from the robot and/or the environment which is used to provide a cost estimate. Due to a lack of data for a given robot, transfer learning approaches are often applied to improve traversability for a given robot using recorded data from similar robot types [33]. The data for evaluating the proposed approaches is typically generated in a controlled environment using simple and unrealistic data samples [10]. For example, the authors of [12] record their data for generating a cost map with the objective of energy efficiency using highly simplified trajectories only, driving only straight lines with one fixed velocity. Similar to that, [11] recorded a predefined pattern over an area with constant velocity and linear speed only to generate the training data for their proposed approach. While both approaches prove their importance in the field of traversability analysis, the outcome is expected to be limited and prone to errors in realistic environments. Our approach is evaluated using pre-recorded locomotion data in a realistic off-road environment, reflecting the traversability capabilities of a robot in a real alpine environment. A general overview of terrain traversability methods for robotic systems is given in [28,30].

3. Method

This section presents the proposed method for computing the traversability of a classified terrain using locomotion data. Therefore, locomotion data is pre-recorded for four robotic systems and processed as described within this section. Earth observation data is used as described in [8] to extract information about the terrain that is used to estimate the traversability on different terrain classes. The general method is composed of four main steps. In the first step, the recorded locomotion data is pre-processed to remove outliers and acquire relevant locomotion features which are linked to their recorded pose to determine the corresponding terrain class. The features are then clustered into bins according to their terrain class, such that aggregated data represents the traversability of a class. A cost function representing a weighted sum of the aggregated locomotion features is used to estimate the traversability costs of specific terrain classes. By interpolating and extrapolating costs for unseen terrain classes, the traversability analysis can be extended onto terrain classes that are not represented in the recorded locomotion data. A general overview of the proposed method is depicted in Fig. 1.

3.1. Data preprocessing

Typically, measurements of various sensors such as Inertial Measurement units (IMU), Global Navigation Satellite Systems (GNSS), or odometry represent the movement of a robot through a given environment. Recorded *locomotion data* for a given sensor can be represented as vector $\mathbf{d}_t^{sensor} \in \mathbb{R}^k$, holding k different measurements at time $t \in \mathbb{R}^+$. Since the recorded measurements in $\mathbf{d}_t^{sensor}$ in general consist of raw sensor readings (e.g. IMU data), it is necessary to process them in order to extract relevant *locomotion features* which can be used to analyze the traversability. Such features can, for example, contain information on the relative movement of the robot (e.g. velocity, acceleration), information on the terrain (e.g. smoothness of the terrain, wheel slippage), or other robot-specific information (e.g. power consumption).

To extract such locomotion features at time t, a function *extract* is used which takes the raw sensor measurements in a time interval $[t-a, t+b]$ as input and provides a vector of M locomotion features such that

$$extract(D^{sensor}) : \mathbb{R}^{k\times|D^{sensor}|} \rightarrow \mathbb{R}^M \tag{1}$$

with

$$D^{sensor} = \{\mathbf{d}_x^{sensor} | t-a \leq x \leq t+b\} \tag{2}$$

and $a, b \in \mathbb{R}^+$ being fixed values for all data points. We denote the output of this function as $\mathbf{f}_t^{sensor} = extract(D^{sensor})$, which represents the locomotion features at time t. Examples of feature extraction within a time interval are acceleration features from known positions or a moving average over the velocity.

In the next step, the extracted feature dataset is filtered manually to remove invalid sequences in which the robot received assistance from a human (e.g. lifting or pushing a robot that got stuck). After that, outlier detection is conducted to remove invalid measurements. Therefore, various filtering techniques can be used. The use of a Gaussian Mixture Model (GMM) has been proven to perform well for univariate as well as multivariate outlier detection (e.g. [34–36]) and can be used for this step. A comparative evaluation of other outlier detection algorithms can be found in [37]. The density of a locomotion feature in a GMM is given by a convex linear combination of multiple components, such that

$$gmm(\mathbf{f}_t^{sensor}) = \sum_{g=1}^{G} \pi_g \phi(\mathbf{f}_t^{sensor} | \mu_g \Sigma_g), \tag{3}$$

where G is the number of components, $\pi_g > 0$ is the mixing proportion over the clusters with $\sum_{g=1}^{G} \pi_g = 1$, and

$$\phi(\mathbf{f}_t^{sensor} | \mu_g, \Sigma_g) = \frac{\exp\left\{-\frac{1}{2}(\mathbf{f}_t^{sensor} - \mu_g)^T \Sigma_g^{-1} (\mathbf{f}_t^{sensor} - \mu_g)\right\}}{\sqrt{(2\pi)^M |\Sigma_g|}} \tag{4}$$

is the Gaussian distribution representing the density of the M-dimensional variable $\mathbf{f}_t^{sensor}$ with mean μ_g and covariance Σ_g. Given a locomotion feature $\mathbf{f}_t^{sensor}$, the GMM calculates the density of the feature on each component. Data points with low density indicate an anomaly and can be discarded. Thus, we can represent the filtered data for a sensor as a collection of all feature vectors that belong to a component with density λ:

$$G^{sensor} = \{\mathbf{f}_t^{sensor} | gmm(\mathbf{f}_t^{sensor}) > \lambda\}. \tag{5}$$

As discussed later in Section 4, a value of $\lambda = 0.01$ was chosen in this work to remove outliers. From this, we define $\mathbf{g}_t^{sensor} \in G^{sensor}$ to be the filtered feature vector for a given *sensor* at time t.

The filtered features are computed based on raw locomotion data which was recorded in different frequencies. To create a dataset with a common sampling rate *freq*, the locomotion features in G^{sensor} need to be sampled every T seconds (with *freq* =

$1/T$), resulting in a sequenced set $S^{sensor} = \{\mathbf{g}_{iT}^{sensor} | i \in \mathbb{N}\}$, holding sampled locomotion features for a sensor with frequency *freq*. Then, all the sampled features from the sensors are merged into a collective set $S^{sampled}$, forming a new feature vector $\mathbf{s}_i^{sampled}$ for sequence i, by merging the sensor features that have the least temporal variation: $\mathbf{s}_i^{sampled} = \langle \mathbf{s}_i^{sensor} \rangle, \forall sensor$. For sensors with sampling frequencies that are not integer multiples of each other, sampling-frequency conversion was applied [38].

3.2. Feature clustering

Clustered locomotion features are used to describe the traversability of the terrain. The sampled features need to be clustered according to the underlying terrain classes at the positions where they were recorded. This information can be retrieved using earth observation data of the recording environment such as orthoimages, land cover information, and height maps. The features in $S^{sampled}$ hold information about the robot's relative movement and are linked to an absolute position within the environment. We therefore define $\mathbf{p}_i$ to be the absolute position of the sampled feature $\mathbf{s}_i^{sampled}$. $\mathbf{p}_i$ is presented in the UTM coordinate frame using patch 33T. Each position $\mathbf{p}_i$ refers to a terrain class derived from earth observation data, describing the terrain at this point. Terrain classes are defined to be multidimensional and thus hold multiple pieces of information for a specific position in the environment. Terrain may be described by the type of surface and its slope, or even consider weather conditions (e.g. dry vs. wet). We define a multidimensional terrain class as a vector of N discrete elements c_i: $T = c_1 \times c_2 \times \cdots \times c_N$ holding terrain information for a given position. Such a vector $\mathbf{t}_i \in T$ thus represents the terrain class for a sampled feature point $\mathbf{s}_i^{sampled}$ at location $\mathbf{p}_i$.

Having linked the terrain classes to the sampled features, one can use data binning to assign every feature to a corresponding collection of terrain class bins, such that $\forall \mathbf{t}_j \in T : bin_j = \{\mathbf{s}_i^{sampled} | \mathbf{t}_j = \mathbf{t}_i\}$. Next, the features which fall into bin_j, are aggregated to create a condensed representation of that bin describing the traversability of the terrain. The aggregated locomotion features are defined as $\mathbf{L}^{\mathbf{t}_j} \in [0, 1]^M$, meaning for every terrain class $\mathbf{t}_j$ there are M aggregated locomotion features describing its traversability. The aggregation of locomotion features can be obtained using statistical parameters derived from the feature set, such as mean or median. Examples can be the mean velocity and smoothness of a particular terrain class. To ensure proper scaling of different feature types we normalize them to the interval [0, 1]. We invert the normalized features where a low value indicates difficult traversability (e.g. velocity). For binning, it is necessary to define the number of minimum samples within a bin to ensure meaningful values. Therefore, only bins with at least $|bin_j| \geq \epsilon$ samples should be considered for cost calculation. Within our experiments, a value of $\epsilon = 6$ was chosen (see Section 4).

3.3. Data collection

To collect enough data points for the individual bins of the terrain classes, it is necessary to prepare the data recording well. Thus, we define a recording experiment as a defined path through an environment in which locomotion data are recorded. Using a random path for data recording is possible, but it poses the risk of not collecting sufficient information for each terrain class and may not be efficient.

To successfully plan the data recording, it is necessary to have a map with a classification of the terrain of that area. Here we use segmented earth observation data which provide information on the land cover type and the steepness of the terrain. In the first step, classes that are completely impassable are identified and excluded (e.g. buildings, lakes, cliffs...).

The recording path through the remaining areas is then planned in a way that it covers all terrain classes sufficiently well. For practical reasons, paths are planned as lap to return to the starting point. To cover the classes well, it might be necessary to plan more than one path because of problems in the general accessibility and connectivity of regions. In particular, we need to incorporate sections that show the limitations of the robotic platform. Anyhow, sections that are easy to traverse need to be recorded as well to get a baseline for the traversability.

The length of the path of the recording has to be chosen in a way, that sufficiently many data points are collected. Moreover, for efficient execution of the recording experiments, the battery run-time of the robots needs to be considered in the planning.

In the data recording procedure, it is important to have means to mark areas that were planned to pass but the robot was not able to traverse. Because having no data in a particular area and for a particular terrain class could either mean that the area was not considered or the robot was not able to handle it. The latter needs to be considered manually in the cost estimation.

3.4. Cost calculation

The aggregated locomotion features from Section 3.2 are used to calculate the integrated cost to traverse a specific terrain. A cost function C can be represented as a function $C(\mathbf{t}_j) : T \rightarrow [0, 1]$ which maps the discrete N-dimensional terrain vector $\mathbf{t}_j \in T$, representing a multidimensional terrain class, into a cost value $c \in [0, 1]$ where a value of 0 represents terrains traversable with minimal costs while a value of 1 represents a terrain is not traversable with a certain robot. The cost to traverse a terrain of type $\mathbf{t}_j$ is the weighted sum of the aggregated locomotion features such that

$$C(\mathbf{t}_j) = \begin{cases} 1, & \text{if } \exists l \in M : L_l^{\mathbf{t}_j} \geq \rho_l \\ F(\mathbf{t}_j), & \text{if } F(\mathbf{t}_j) < 1 \\ 1, & \text{otherwise} \end{cases} \tag{6}$$

with

$$F(\mathbf{t}_j) = \sum_{l}^{M} w_l \cdot L_l^{\mathbf{t}_j} \tag{7}$$

where $w_l \in \mathbb{R}$ is the weight factor for the lth aggregated locomotion feature $L_l \in [0, 1]$. The first condition in Eq. (6) states that terrain classes are not traversable if one of the aggregated locomotion features exceeds its safety limit $\rho_l \in [0, 1]$. The second condition sums the weighted aggregated locomotion features and the third condition cuts the total cost to a maximum of 1 if the weighted sum exceeds this value.

3.5. Cost interpolation

Since the recorded locomotion data typically cannot cover all possible multidimensional terrain classes, the resulting cost table will be sparse. Missing locomotion features can be extended by either manually defining costs for missing terrain classes with expert knowledge or by interpolating existing costs derived using the methods presented in Section 3.4.

Costs can be interpolated (extrapolated) along one or more dimensions of the terrain class if the dimensions are of ordinal nature and there are at least two cost estimates within the terrain dimensions. An example of an ordinal dimension is the slope profile of the environment, which is divided into ordered discrete classes. To interpolate terrain costs for a given terrain class $\mathbf{t}_i$ according to a set of ordinal dimensions $\mathcal{N}$, we represent the data

as a relation R [39]. First, we perform a selection based on the values in the dimensions of t_i disjoint from that in $\mathcal{N}$ and the cost dimension, $R' = \sigma_{\pi_{\{R_S \setminus \{\mathcal{N}, cost\}\}}(R) = \mathbf{t}_i}(R)$. To allow interpolation of the terrain selection, at least two disjoint values are needed for each dimension in $\mathcal{N}$ after the selection. Using linear interpolation, for each terrain vector $\mathbf{t}_i$ with unknown costs which lies between two cost estimates $C(\mathbf{t}_j^{R'})$, and $C(\mathbf{t}_k^{R'})$ from R', $j < k$ representing the order along $\mathcal{N}$, the cost can be estimated as

$$C(\mathbf{t}_i^{R'}) = C(\mathbf{t}_j^{R'}) + \frac{C(\mathbf{t}_k^{R'}) - C(\mathbf{t}_j^{R'})}{k - j}(i - j). \tag{8}$$

The same equation can be used for extrapolating the costs outside of terrain classes if no other cost estimates exist. When extrapolating, however, it must be ensured that terrain classes without a cost estimation are basically traversable. Otherwise, terrain may be extrapolated which does not contain any measurements due to its inaccessibility.

Additionally, an expert can use the basis of the interpolated costs to further fill in missing information for all the other terrain classes which could not be interpolated on the aggregated locomotion features. Another option is also to improve the cost interpolation for unknown terrain classes using machine learning approaches, such as ordinal regression [40].

4. Experiments & evaluation

To evaluate the proposed pipeline for traversability analysis, experiments were conducted in an off-road alpine environment to collect locomotion data from four different robot types. This data was later processed with the pipeline and the results were investigated. First, this section presents the setup and terrain for the data recording. Second, the process of the pipeline to extract traversability costs is presented in detail and evaluated in relation to the experimental environment.

4.1. Experimental setup

One of the objectives of the work is to show that the traversability in off-road terrain differs for different types of platforms, and thus an individual assessment of traversability is necessary. For this purpose, robots were selected that differ from each other as much as possible in order to highlight the differences in traversability. Therefore, the traversability analysis was conducted using four different mobile robots which differ in various ways, such as locomotion type, size, or weight, and thus have completely different capabilities for off-road navigation. The used robots were a tracked robot (Rovo3) from Mattro,[1] a wheeled robot with skid-steer drive (Husky) from Clearpath Robotics,[2] a legged robot (Spot) from Boston Dynamics,[3] and a wheeled robot with double Ackermann steering (Mercator), developed by the Graz University of Technology (TU Graz) [41]. Fig. 2 depicts the used robots. Table 1 provides an overview of the robots' specifications. The experiments were conducted on the Seetaler Alps, an alpine off-road environment and military training ground in Austria. Data recording was conducted in two main areas, as shown in Fig. 3a. To establish comparability among the robot systems when recording data, a universal mounting frame was developed that contains the sensor setup and can be mounted on each robot (see Fig. 4). The frame holds various time-synchronized sensors to record the locomotion of the robot, namely a GNSS Sensor with RTK from geo-konzept to allow precise position measurements, an Xsens MTi-G-710 IMU, and a Stereolabs ZED2 stereo camera. The frame is mounted centered on each robot with the stereo camera pointing forward. The frame is also connected to the robot to record internal status information, such as wheel odometry. As some robots have only a small battery to use, an external power supply for the frame is mounted to extend the robot's battery life.

Fig. 2. The four robots used for data recording. From left to right: Mattro Rovo3, Clearpath Robotics Husky, Boston Dynamics Spot, TU Graz Mercator.

Table 1
Specifications of the four robots used for data recording.

	Rovo3	Husky	Spot	Mercator
Dimensions (L×W×H in m)	1.2 × 1.2 × 0.5	1.0 × 0.7 × 0.4	1.1 × 0.5 × 0.7	1.8 × 1.2 × 1.2
Weight (kg)	295	50	32	550
Ground clearance (m)	0.15	0.13	0.3	0.12
Max speed (m/s)	4.15	1.0	1.6	3.0
Locomotion Type	Skid Steering (Tracked)	Skid Steering (Wheeled)	Legged	Double Ackerman (Wheeled)

4.2. Dataset

To conduct the traversability analysis and extract costs for various off-road terrains, an appropriate dataset needs to be provided. This section presents the used environment and data recording process. In order to allow the replication of our experiments and to foster further research on off-road terrain analysis the recorded dataset is publicly available.[4]

The raw locomotion data is recorded with the universal frame mounted on the four robots as described in Section 4.1. The robots were controlled manually by an operator who followed the robot. The maximum speed of all four robots was limited to 2 m/s to allow the operator to keep up while controlling the robot. Position data and robot internal data was recorded with a frequency of 20 Hz while IMU measurements were recorded at 100 Hz.

The environment was segmented using two terrain dimensions. One dimension represents the height profile of the environment while the other dimension represents the land cover class (see Figs. 3b and 3c). The height profile is segmented into seven discrete classes $[-3, 3]$ representing a range of slopes. The class intervals are given in ° and defined as $-3 \sim [-45, -15)$, $-2 \sim [-15, -10)$, $-1 \sim [-10, -3)$, $0 \sim [-3, 3)$, $1 \sim [3, 10)$, $2 \sim [10, 15)$ and $3 \sim [15, 45]$. Depending on the motion of the robot (up or down), the positive or negative slope is used to determine the corresponding class. For land cover classes, a set of 17 classes were identified and labeled as shown in Fig. 3c.

The paths for the data recording were derived following the process from Section 3.3, coming up with two routes within the environment, as visualized in Fig. 3. Route 1 traverses an area (Testarea 1) containing forests, open vegetation, and roads. Route 2 (Testarea 2) is a highly alpine environment above the tree line and provides information on the alpine meadows and various other vegetation types. The two routes, which were traversed

[1] www.hawe.com/products/robot-platform/ (accessed on 07.11.2022)

[2] www.clearpathrobotics.com/husky-unmanned-ground-vehicle-robot/ (accessed on 07.11.2022)

[3] www.bostondynamics.com/products/spot (accessed on 07.11.2022)

[4] https://robonav.ist.tugraz.at/data/

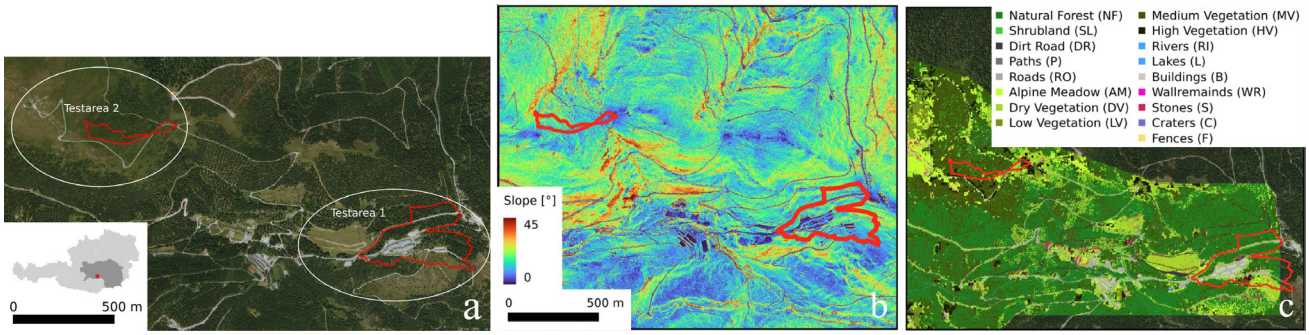

Fig. 3. Overview of the test environment. (a) orthophotographic overview of the experimental environment. The data was recorded on the Seetaler Alps, an alpine environment in Austria. (b) environment segmented into multidimensional terrain classes containing slope information, and (c) segmented environment into 17 land cover classes. Red lines indicate the planned routes for the two test runs, which were recorded for every robot. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

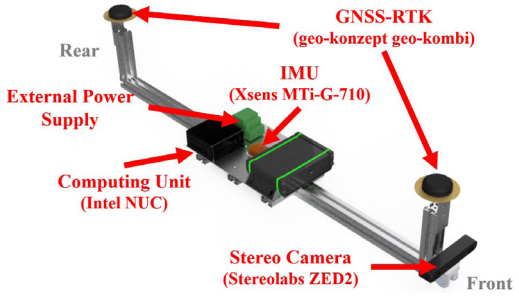

Fig. 4. The sensor setup for data recording. While every robot provided its kinematic information, the sensor setup on the mounting frame was the same for all robot types.

Table 2
Terrain class distribution of Route 1.

	%	Land cover									
		NF	SL	DR	P	RO	AM	DV	LV	MV	HV
Slope Class	−3 [−45, −15)	6.85	0.39	1.24	0.40	2.28	0.66	0.00	1.21	0.00	2.56
	−2 [−15, −10)	2.87	0.17	1.28	0.19	1.42	0.43	0,04	2.02	0.00	2.09
	−1 [−10, −3)	1.25	0.28	4.43	0.31	6.19	0.31	0,02	2.34	0.00	1.03
	0 [−3, 3)	0.22	0.13	1.89	0.11	15.93	0.07	0.00	0.49	0.00	0.10
	1 [3, 10)	2.60	0.11	3.21	0.22	3.54	0.22	0.00	2.67	0.00	0.98
	2 [10, 15)	3.42	0.19	1.16	0.17	0.55	0.25	0.00	2.12	0.00	1.49
	3 [15, 45]	9.38	0.32	0.68	0.24	1.30	0.35	0.00	1.30	0.00	2.25

with every robot, were planned according to land cover and slope variety as well as their accessibility. The land cover types have been determined using remote sensing with aerial images. For planning, the distribution of land cover types within the test areas was used to determine the most prominent classes; natural forest (21%) and high vegetation (18%) made up the majority of the terrain, followed by alpine meadows (13%) and low vegetation (13%). Routes were primarily designed to incorporate these classes. The mean slope of the full test area is 20.32°, with 50% being slopes larger than 15°.

The route in Testarea 1 was intended to contain natural forests with high slopes ($> 15°$) to test the limits of the robotic platforms. It features also a section along a paved road to have a baseline comparison of where traversing should be possible for every robot. Part of the route also goes through low vegetation that was used as a pasture at the time of data collection. The second route in Testarea 2 was primarily chosen because it contains up to 37% high vegetation and 20% alpine meadows with high slopes; 20% downhill and 19% uphill slopes ($> 15°$). A general overview of the slope and land cover distribution of the planned routes is given in Table 2 and Table 3. The abbreviations for the land cover classes are taken from Fig. 3c.

Table 3
Terrain class distribution of Route 2.

	%	Land cover									
		NF	SL	DR	P	RO	AM	DV	LV	MV	HV
Slope Class	−3 [−45, −15)	0.00	1.49	0.00	0.82	0.14	5.62	2.94	0.14	1.68	6.24
	−2 [−15, −10)	0.00	1.14	0.00	2.18	0.33	5.38	3.45	0.55	1.84	6.17
	−1 [−10, −3)	0.00	0.46	0.03	1.19	1.83	1.27	2.58	1.18	0.88	2.62
	0 [−3, 3)	0.00	0.00	0.00	0.06	0.09	0.10	0.96	0.08	0.11	0.08
	1 [3, 10)	0.00	0.33	0.03	0.10	2.73	1.58	1.48	0.65	1.67	1.79
	2 [10, 15)	0.00	0.91	0.03	0.19	1.00	2.55	0.60	0.17	3.16	6.02
	3 [15, 45]	0.00	2.15	0.00	0.11	0.06	2.06	0.65	0.11	2.87	13.35

As the robots are not able to follow the given path exactly or not able to access parts of the route, the number of recorded samples and the distribution of the land cover and slope classes slightly differ. Table 4 provides a statistical overview of the different recorded datasets for all four robots. This also shows several differences in the recorded lengths, especially for Spot and Mercator. In Route 1, Spot was only able to traverse a small part of the planned route due to overly difficult terrain, resulting in a shortened data recording. For Mercator, part of Route 1 was blocked by a fence that could not be bypassed. Thus, part of the route had to be backtracked and is therefore covered twice. Also, the terrain in Route 2 was only passable for Mercator to a limited extent, which resulted in a shorter route being recorded here.

4.3. Traversability cost analysis

The processing of the raw locomotion data from the recorded dataset was done by extracting the raw sensor readings from a recorded ROS bagfile from the GNSS (timestamp, latitude, longitude, height above sea level, orientation) and IMU (timestamp, linear accelerations, angular velocities). This data was then used two extract two features which were then resampled with a frequency of 1 Hz and time-synchronized as described in Section 3. First, the velocity using the GNSS profile of the robot on a given terrain, and secondly, the recording of the general smoothness of the terrain which was derived using IMU measurements. To derive the smoothness, a movement characterization method called spectral arc length (SAL) was used on the angular velocity measurements of the IMU [42]. Therefore, the arc length of the Fourier magnitude spectrum was measured for each angular velocity (roll, pitch, yaw) within a fixed frequency range of 20–35 Hz, as determined in [43], and averaged over all three velocities.

To improve results, outlier removal was conducted using GMM shown in Eq. (3) on one component ($G = 1$) with a density threshold of $\lambda = 0.01$. Additionally, sequences in which operators had to intervene were filtered out manually using the recordings of the stereo camera.

Table 4
Statistics of the recorded data on all four robots.

	Rovo3		Husky		Spot		Mercator	
	Route 1	Route 2	Route 1	Route 2	Route 1	Route 2	Route 1	Route 2
Distance (m)	2 334	2 045	2 792	1 493	1 370	2 462	3 666	1 025
# IMU Measurements	279 929	276 296	455 880	276 106	219 076	351 806	434 483	138 798
# GNSS Measurements	55 778	53 804	91 073	55 220	43 675	70 360	86 895	27 760
# Camera Images	2329	2307	3 566	2 185	1 872	2 946	3 422	1 065
Slope (%)	Route 1	Route 2	Route 1	Route 2	Route 1	Route 2	Route 1	Route 2
−3 [−45, −15)	6.24	13.73	12.78	10.43	1.92	0.90	7.81	7.11
−2 [−15, −10)	8.04	19.55	8.18	20.39	11.04	11.80	8.81	24.98
−1 [−10, −3)	35.49	9.09	26.80	12.96	36.19	36.51	35.15	18.07
0 [−3, 3)	10.92	2.04	10.56	4.13	17.39	3.66	12.89	4.24
1 [3, 10)	12.73	11.69	14.71	15.36	25.25	15.06	23.75	16.09
2 [10, 15)	8.58	14.91	10.97	16.82	6.12	20.96	6.98	23.20
3 [15, 45]	17.99	28.99	15.99	19.92	2.09	11.12	4.61	6.31
Land Cover (%)	Route 1	Route 2	Route 1	Route 2	Route 1	Route 2	Route 1	Route 2
Natural Forest (NF)	23.55	0.00	28.62	0.00	8.79	0.00	7.17	0.00
Shrubland (SL)	0.00	10.42	0.00	3.24	4.06	5.77	1.26	6.61
Dirt Road (DR)	3.95	0.00	12.14	0.00	10.68	0.00	30.88	0.00
Paths (P)	0.00	8.52	0.00	21.23	0.00	5.44	0.00	17.37
Roads (RO)	37.59	8.58	27.84	5.40	69.52	62.81	39.53	13.82
Alpine Meadow (AM)	10.53	14.80	0.00	2.54	0.00	12.96	0.00	15.10
Dry Vegetation (DV)	0.00	3.57	0.00	9.35	0.00	3.97	0.00	7.60
Low Vegetation (LV)	15.26	0.00	18.16	3.43	6.95	0.00	11.51	4.24
Medium Vegetation (MV)	0.00	11.23	0.00	1.74	0.00	6.37	0.00	30.70
High Vegetation (HV)	9.12	42.88	13.24	53.08	0.00	2.68	9.65	4.54

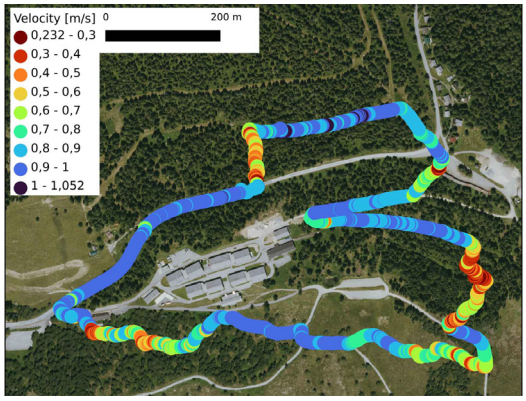

Fig. 5. Example of the mapping of processed data (velocity profile) on GNSS coordinates traversed by Husky on route 1.

The resulting features were geo-located using the GNSS data and assigned to the corresponding terrain classes, as exemplarily shown for the velocity profile of Husky in Fig. 5. In the visualization, it can already be seen, that the environment impacts the velocity driven significantly.

Before clustering all extracted features into their terrain classes, up to nine path segments of a given terrain structure with a length of up to 40 meters were randomly excluded from the next steps and used for validation in Section 4.4. The remaining features were clustered into their corresponding terrain bins and aggregated into two values: the average driven velocity and the average smoothness of the terrain. To represent these values in [0, 1], the two aggregated features were normalized using feature scaling [44] based on the maximum feature value of each robot. For aggregating the locomotion features, only bins were selected which at least contain 6 data samples ($|bin_j| \geq 6$).

An overview of aggregated locomotion features for different terrain classes on Husky which were extracted from the recorded dataset is shown in Table 5. The first row in each cell of the table represents the aggregated feature value and the standard deviation (σ) for velocity, the second row represents the smoothness of the terrain. The horizontal axis defines the land cover classes as defined in Fig. 3c, the vertical axis shows the slope classes as defined in Section 4.3. While terrain classes that are not traversable, such as lakes (L) or buildings (B), are not represented in the table, we see for instance that not sufficient data for a flat alpine meadow (AM) was recorded and is thus left blank. The data shows that the traversability of Husky in terrain classes such as natural forests (NF) or shrubland (SL) has high velocity values in all slope classes resulting in an overall high cost estimate. The aggregated features on paved ways (DR, P, RO) on the other hand show overall low values for velocity, indicating good traversability in terms of speed. The results of the other robots can be found in the supplementary material.

Using the cost function from Eq. (7), the traversability of a specific terrain can be computed. The weights for our cost function can be set to either favor fast routes (velocity) or smooth routes (smoothness). The selection of the weights usually depends on the demands on the robot and its application. For example, a robot may be required to get to the destination as quickly as possible without putting any importance on the smoothness of the surface. In other scenarios (e.g. transport of sensitive goods), it may be more important to reach the destination as smoothly as possible, in which case speed plays only a minor role. For our experiments, a combined weighting approach was selected which slightly favors a faster route: $w_0 = 1.0$ (velocity), $w_1 = 0.5$ (smoothness). To make the four robot systems comparable, the costs have been calculated for all of them using the same weighting factors. To allow safer navigation, terrain classes that have a single feature greater than 0.95 are set to be non-traversable ($\rho = 0.95$ from Eq. (6)). After interpolating the cost values over the ordinal slope information as described in Section 3.5, additional cost values could be identified. Blocked land cover classes such as buildings, lakes, or fences were manually set at the highest cost to avoid planning through such environments. Table 6 shows

Table 5
Table showing the two aggregated locomotion features (velocity and smoothness) and their standard deviation (σ) for each terrain class on the Husky data.

Slope Class	Feature	NF $\mathbf{t}_i^j$ L_i^j	NF σ	SL $\mathbf{t}_i^j$ L_i^j	SL σ	DR $\mathbf{t}_i^j$ L_i^j	DR σ	P $\mathbf{t}_i^j$ L_i^j	P σ	RO $\mathbf{t}_i^j$ L_i^j	RO σ	AM $\mathbf{t}_i^j$ L_i^j	AM σ	DV $\mathbf{t}_i^j$ L_i^j	DV σ	LV $\mathbf{t}_i^j$ L_i^j	LV σ	MV $\mathbf{t}_i^j$ L_i^j	MV σ	HV $\mathbf{t}_i^j$ L_i^j	HV σ	Overall $\mathbf{t}_i^j$ L_i^j	Overall σ
Land cover class																							
−3	vel.	0.74	*(0.17)*	0.39	*(0.18)*	0.17	*(0.15)*	0.08	*(0.11)*	0.27	*(0.05)*	–	-	0.35	*(0.23)*	–	-	0.55	*(0.21)*	–	-	**0.37**	**(0.16)**
	smth.	0.19	*(0.13)*	0.29	*(0.10)*	0.38	*(0.14)*	0.65	*(0.15)*	0.58	*(0.13)*	–	-	0.41	*(0.14)*	–	-	0.28	*(0.13)*	–	-	**0.40**	**(0.13)**
−2	vel.	0.68	*(0.18)*	0.71	*(0.21)*	0.16	*(0.14)*	0.05	*(0.08)*	0.21	*(0.09)*	0.35	*(0.24)*	0.28	*(0.26)*	–	-	0.57	*(0.22)*	0.51	*(0.19)*	**0.39**	**(0.18)**
	smth.	0.22	*(0.12)*	0.17	*(0.09)*	0.33	*(0.10)*	0.61	*(0.14)*	0.54	*(0.12)*	0.72	*(0.16)*	0.50	*(0.16)*	–	-	0.30	*(0.13)*	0.48	*(0.13)*	**0.43**	**(0.13)**
−1	vel.	0.70	*(0.17)*	–	-	0.14	*(0.09)*	0.04	*(0.06)*	0.18	*(0.10)*	–	-	0.32	*(0.16)*	0.05	*(0.16)*	0.37	*(0.18)*	0.23	*(0.18)*	**0.25**	**(0.14)**
	smth.	0.21	*(0.13)*	–	-	0.37	*(0.12)*	0.57	*(0.13)*	0.47	*(0.12)*	–	-	0.45	*(0.12)*	0.47	*(0.14)*	0.39	*(0.12)*	0.39	*(0.15)*	**0.42**	**(0.13)**
0	vel.	0.38	*(0.14)*	–	-	0.18	*(0.11)*	–	-	0.23	*(0.11)*	–	-	0.30	*(0.24)*	–	-	0.44	*(0.14)*	0.12	*(0.15)*	**0.27**	**(0.15)**
	smth.	0.37	*(0.12)*	–	-	0.34	*(0.10)*	–	-	0.44	*(0.11)*	–	-	0.38	*(0.13)*	–	-	0.34	*(0.09)*	0.40	*(0.12)*	**0.38**	**(0.11)**
1	vel.	0.74	*(0.16)*	0.50	*(0.18)*	0.22	*(0.12)*	–	-	0.20	*(0.08)*	0.22	*(0.16)*	0.36	*(0.19)*	0.18	*(0.14)*	0.41	*(0.19)*	0.18	*(0.16)*	**0.34**	**(0.15)**
	smth.	0.20	*(0.13)*	0.33	*(0.12)*	0.41	*(0.13)*	–	-	0.42	*(0.10)*	0.52	*(0.14)*	0.41	*(0.11)*	0.47	*(0.12)*	0.35	*(0.13)*	0.40	*(0.14)*	**0.39**	**(0.13)**
2	vel.	0.62	*(0.16)*	0.55	*(0.17)*	0.30	*(0.17)*	0.28	*(0.16)*	0.22	*(0.07)*	0.26	*(0.19)*	0.34	*(0.28)*	0.25	*(0.10)*	0.45	*(0.18)*	0.57	*(0.15)*	**0.38**	**(0.16)**
	smth.	0.27	*(0.12)*	0.32	*(0.11)*	0.46	*(0.13)*	0.58	*(0.15)*	0.45	*(0.10)*	0.55	*(0.11)*	0.45	*(0.15)*	0.41	*(0.11)*	0.33	*(0.13)*	0.35	*(0.12)*	**0.42**	**(0.12)**
3	vel.	0.72	*(0.15)*	0.56	*(0.19)*	–	-	–	-	0.22	*(0.06)*	0.36	*(0.27)*	0.26	*(0.26)*	–	-	0.50	*(0.22)*	–	-	**0.44**	**(0.19)**
	smth.	0.22	*(0.13)*	0.26	*(0.08)*	–	-	–	-	0.47	*(0.08)*	0.43	*(0.13)*	0.48	*(0.14)*	–	-	0.28	*(0.11)*	–	-	**0.36**	**(0.11)**
Overall	vel.	**0.65**	**(0.16)**	**0.55**	**(0.19)**	**0.19**	**(0.14)**	**0.11**	**(0.10)**	**0.22**	**(0.08)**	**0.30**	**(0.22)**	**0.32**	**(0.23)**	**0.16**	**(0.13)**	**0.47**	**(0.19)**	**0.32**	**(0.16)**	**0.33**	**(0.16)**
	smth.	**0.24**	**(0.13)**	**0.27**	**(0.10)**	**0.38**	**(0.12)**	**0.60**	**(0.14)**	**0.48**	**(0.11)**	**0.55**	**(0.14)**	**0.44**	**(0.14)**	**0.45**	**(0.12)**	**0.32**	**(0.12)**	**0.40**	**(0.13)**	**0.41**	**(0.12)**

Table 6
Cost table after interpolation from Husky data.

Slope Class	NF	SL	DR	P	RO	AM	DV	LV	MV	HV	RI	L	B	WR	S	C	F
	Land cover class																
-3	1.00	0.61	0.37	0.48	0.70	1.00	0.66	0.00	0.82	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
-2	0.96	0.96	0.30	0.40	0.57	1.00	0.64	0.13	0.87	0.94	1.00	1.00	1.00	1.00	1.00	1.00	1.00
-1	0.98	0.87	0.32	0.36	0.46	0.74	0.65	0.28	0.66	0.47	1.00	1.00	1.00	1.00	1.00	1.00	1.00
0	0.66	0.73	0.34	0.32	0.50	0.47	0.55	0.42	0.71	0.31	1.00	1.00	1.00	1.00	1.00	1.00	1.00
1	1.00	0.80	0.48	0.52	0.44	0.57	0.67	0.47	0.69	0.40	1.00	1.00	1.00	1.00	1.00	1.00	1.00
2	0.92	0.86	0.63	0.71	0.50	0.66	0.67	0.51	0.72	0.92	1.00	1.00	1.00	1.00	1.00	1.00	1.00
3	1.00	0.82	0.78	0.91	0.53	0.69	0.59	0.56	0.76	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00

Costs 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0

the final traversability costs of the terrain classes for the data recorded on Husky. The costs for the terrains computed for Husky are represented by the colors in Table 6. Green indicates low costs (0), while red color indicates high costs (1). Here we also see that we recover a reasonable cost value for the flat alpine meadow because the robot is able to traverse a positively and negatively inclined meadow.

Finally, the cost table can be used for creating a cost map and to plan routes through the environment. Fig. 6 shows the resulting cost maps of the four robot types which were used in the experiment. Since the cost is directional due to the elevation profile, not all costs can be represented as a graphic. Therefore, the slope costs in that graphic are depicted by the maximum positive slope to give a representative visualization. For path planning, the gradient on a certain terrain depends on the movement of the robot and has to be considered accordingly. In the visualization, it can be seen that Rovo3 (Fig. 6a) has little trouble in the alpine meadow and a rather uniform color, indicating only minor differences for the terrain in terms of navigation. Husky (Fig. 6b) performs well in the alpine meadow as well, and also does seemingly better on roads than on harsher terrains. Spot (Fig. 6c) showed the most issues in rough terrain and prefers to stay on paved roads. Mercator (Fig. 6d) too prefers roads and is unable to drive in natural forests (due to its size). However, it performs better than Spot in areas with high vegetation.

4.4. Cost validation

To validate the cost estimates which were derived from the proposed pipeline, several path segments which were excluded from the cost estimation in Section 4.3 were used. Path segments were randomly extracted from the data as consecutive poses prior to cost estimation, with the requirement that they correspond to the required terrain classes, as described in Table 7. Thus, path segments were selected to represent different terrain class variations that occurred during the training process in different complexity. A path segment holds a set of consecutive locations $P = \{\mathbf{p}_i\}$ which can be used to determine the corresponding terrain class $\mathbf{t}_i$ at time i as described in Section 3.2. The resulting path segments resemble a variety of segments ranging from a single class with no slope, over single/multiple classes with a positive or negative slope, to a representative set for vegetation and paved ways in general with various slopes for which consecutive recordings that traverse at least two land cover classes of paved ways (RO, DR, P) or vegetation (LV, MV, HV) was used to represent the driving capabilities in the representative classes. The definition of a representative set for vegetation and paved ways was chosen, as a consecutive recording of all the set classes does not exist for all robot types.

Table 7 gives an overview of the cost predictions on the path segments. The table shows how long an extracted path segment is for a given terrain configuration and presents the predicted and actual costs. The actual and predicted costs C_x for a path segment P is the sum of the actual/predicted cost $C_x(\mathbf{t}_i)$ at point $\mathbf{p}_i$, multiplied by the intermediate distance $d_i = \sqrt{(\mathbf{p}_{i-1} - \mathbf{p}_i)^2}$

$$C_x = \sum_{i=1}^{|P|} C_x(\mathbf{t}_i) \cdot d_i, \tag{9}$$

where $x \in \{a, p\}$ can either be the actual cost (a) or the prediction (p). The relative error e is given in % and is calculated as

$$e = \frac{|C_a - C_p|}{C_a} \cdot 100. \tag{10}$$

For some robot types, a specific (continuous) recording of a desired path does not exist and is therefore denoted with "-". Results show a prediction error of 20–25% on all samples combined for each robot system, resulting in an overall validation error of 21.75%. When evaluating each data sample on all robot

Table 7
Validation results on continuous path segments which were excluded from the cost estimation process.

Data sample description	Rovo3				Husky				Spot				Mercator				Overall error (e)
	Length (m)	Actual (C_a)	Prediction (C_p)	Error (e)	Length (m)	Actual (C_a)	Prediction (C_p)	Error (e)	Length (m)	Actual (C_a)	Prediction (C_p)	Error (e)	Length (m)	Actual (C_a)	Prediction (C_p)	Error (e)	
NF, no slope	5.08	3.23	5.08	36.42	5.08	3.65	3.34	9.28	–	–	–	–	–	–	–	–	**22.85**
DR, negative slope	6.87	5.01	4.28	17.06	5.56	4.09	1.76	132.39	5.16	3.42	5.16	33.72	7.45	3.28	6.31	48.02	**55.31**
RO, positive slope	6.94	5.96	5.31	12.24	5.43	2.86	2.66	7.52	5.61	3.93	4.00	1.75	5.53	5.31	4.86	9.26	**7.94**
LV, various slopes	18.31	9.72	13.41	27.52	5.46	2.04	1.80	13.33	6.93	3.04	3.16	3.80	6.21	4.10	5.47	25.05	**20.55**
AM, RO, positive slope	10.74	8.84	8.86	0.23	19.65	15.08	12.52	20.45	10.78	7.96	5.46	45.79	10.29	7.69	8.02	4.11	**18.27**
AM, RO, negative slope	13.57	9.61	11.21	14.27	12.25	9.20	6.45	42.64	–	–	–	–	9.76	6.71	8.47	20.78	**25.82**
RO,SL,LV, various slopes	–	–	–	–	27.62	15.41	13.73	12.24	20.47	9.84	17.03	42.22	15.57	11.51	13.65	15.68	**22.72**
Ways, various slopes	20.63	15.70	17.57	10.64	11.63	8.31	6.89	20.61	19.71	15.49	12.14	27.59	13.80	9.06	10.68	15.17	**18.43**
Vegetations, various slopes	35.25	19.12	27.45	30.35	39.17	29.79	24.59	21.15	15.05	10.99	10.35	6.18	18.16	14.04	10.53	33.33	**24.12**
All Samples Combined	**117.39**	**77.19**	**93.17**	**20.24**	**131.85**	**90.43**	**73.74**	**24.47**	**83.71**	**57.9**	**62.38**	**19.46**	**86.77**	**61.7**	**67.99**	**21.90**	**21.75**

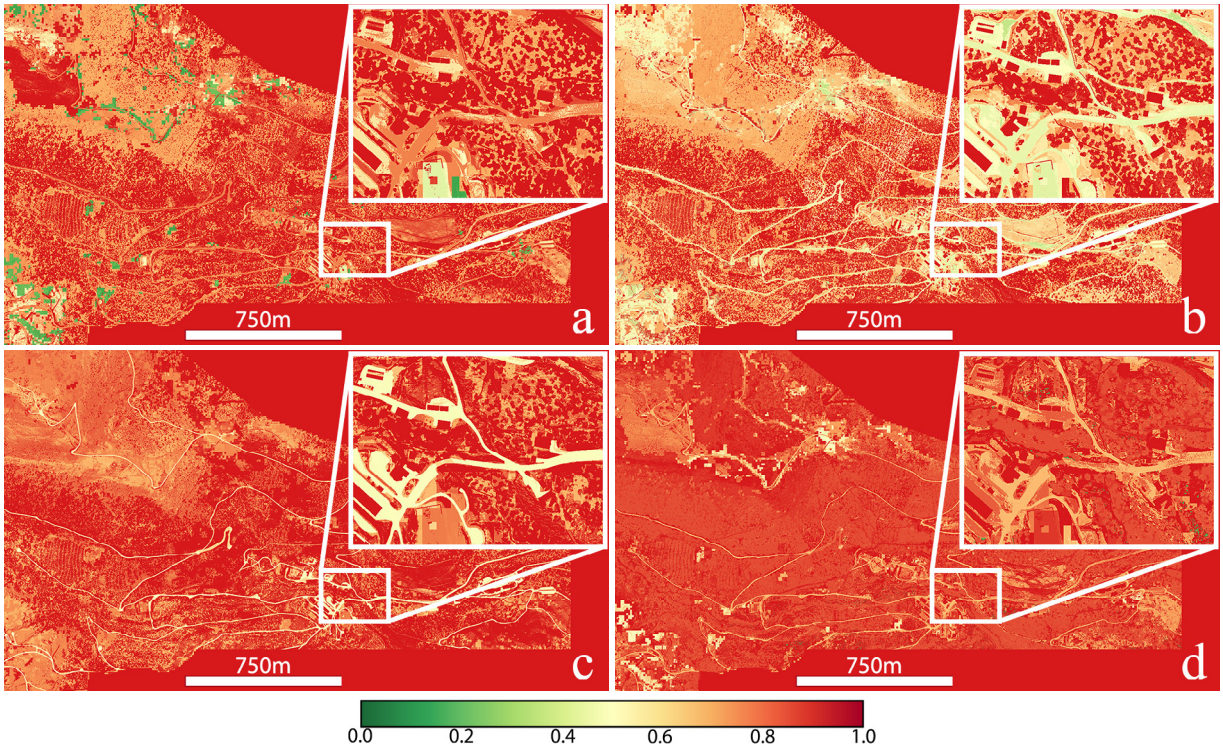

Fig. 6. Representative cost maps of the four different robot types holding the maximum cost value from terrain classes with positive slope. (a) Rovo3 (b) Husky (c) Spot (d) Mercator.

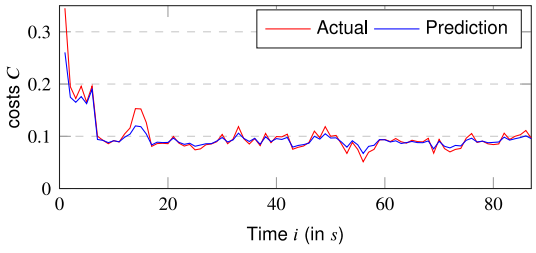

Fig. 7. Plotted costs for the validation of "AM, RO, positive slope" on Rovo3. Due to the relatively small error of 0.23%, the predicted costs overlap nicely with the actual costs indicating a good cost estimate for the given classes.

systems (last column), the error also lies around 20%, with the exception of "DR, negative slope" (55.31%) and "RO, positive slope" (7.94%). Figs. 7 and 8 additionally plot the actual and predicted costs of two exemplary path segments over the distance.

Fig. 7 shows the path segment with the smallest error, recorded with Rovo3 on AM,RO land covers with positive slopes. Here, it can be seen, that the cost estimate fully reflects the robot's capabilities on the given terrain.

In contrast, Fig. 8 shows the cost prediction of Husky on the path "DR, negative slope" which has the biggest error of 132.39%. Based on the plot, one can see that the actual costs vary a lot within the terrain class, indicating that it is not possible to represent negative "DR" classes with a single cost estimate. Also, the cost estimate for the class seems to be too low for the extracted validation sample, indicating that a path segment was (randomly) selected which does not fully reflect the estimated costs from the dataset.

4.5. Path planning

To investigate the usefulness and impact of the generated cost maps concerning path planning, routes through the environment were planned using the maps and later analyzed. For this, the widely applied A* search algorithm [45] was used to plan routes with pre-defined start and goal positions using the computed cost maps. To consider the direction of slopes in path planning, layered cost maps can be used [46] in which each layer represents the costs for a specific slope direction. Based on the planner's direction of expansion, the corresponding layer can be selected. An example route for all four robot types can be seen in Fig. 9. From this visualization it can be seen, that Rovo3 (Fig. 9a) is

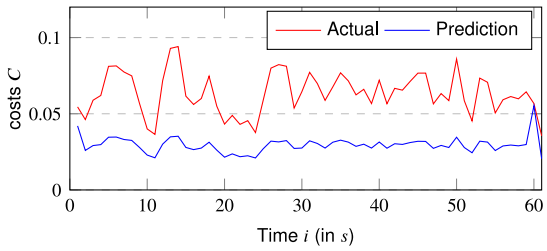

Fig. 8. Plotted costs for the validation of "DR, negative slope" on Husky. One can see, that the error of 132.39% arises due to a high variance in the actual costs in the given terrain class, which cannot be reflected by the computed cost estimates. Also, the cost estimate for the class seems to be too low for the extracted validation sample.

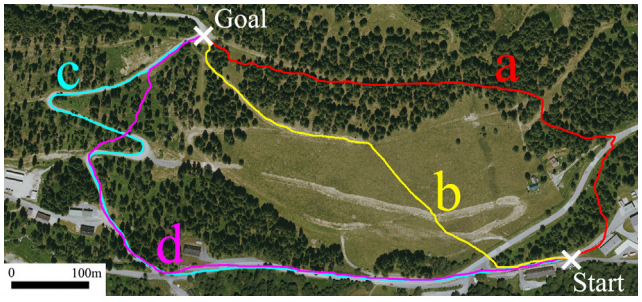

Fig. 9. Path planning results using A* for (a) Rovo3, (b) Husky, (c) Spot, (d) and Mercator. The paths indicate the planned route through the terrain using the computed cost estimates.

capable of going through the natural forest on a dirt road and thus is able to follow a straight shorter path. Husky (Fig. 9b) has little trouble within all different types of terrain and is also able to traverse steep slopes, letting it plan rather directly through the environment. Spot (Fig. 9c) and Mercator (Fig. 9d) have more trouble navigating in harsh environments and prefer to stay on dirt roads. However, while Spot does not allow any deviation from the road, Mercator is able to take some shortcuts.

To validate the resulting paths, a qualitative evaluation was conducted with three experts each who were either involved in the development of the robot or have used the platform intensively in the field. During the evaluation, the experts were asked about the robot's capabilities in off-road terrain and how they would rate its navigation capability. They were then shown an orthographic satellite image, a height image, as well as the land cover information of the same area as shown in Fig. 9 and asked to draw the optimal route which they think the robot could traverse, given the same start and end positions. The resulting paths from the experts are visualized in Fig. 10. One can see, that for Husky (Fig. 10b) and Spot (Fig. 10c), the experts had a similar opinion on the optimal path of the robot. For Mercator (Fig. 10d), Expert 1 had a similar route in mind as computed, while Expert 2 would have sent the robot a completely different way. Also, for Rovo3 (Fig. 10a), all three experts expected a different route to be optimal. When asked why, they stated that they thought that the open area would be easier to traverse for the robot. Furthermore, the experts often did not see details that would have led to a preferred route. Following on the example of Mercator, the experts had difficulty recognizing the path through the forest as such and therefore rated it as worse than the open area. To compare the similarity of the computed and expert paths, Path Similarity Analysis (PSA) [47] was conducted using the discrete Frèchet distance [48] as metric. The metric indicates an identical path if the distance is zero, while large distances indicate dissimilarity. Table 8 shows the results of the path similarity

Table 8
Path similarity analysis on the expert evaluation results.

Robot	Computed Path	Expert 1		Expert 2		Expert 3	
	Length (m)	Length (m)	Fréchet (m)	Length (m)	Fréchet (m)	Length (m)	Fréchet (m)
Rovo3	765.03	624.01	213.67	636.18	235.84	750.68	204.63
Husky	612.94	642.80	31.54	767.00	89.21	623.79	52.06
Spot	1152.80	1185.90	9.73	1184.04	12.49	1164.56	12.19
Mercator	955.01	1179.43	106.64	962.74	194.59	1170.49	184.19

Table 9
Costs for the experts' and computed paths, based on the derived cost maps of each robot.

Robot	Computed Path	Expert 1		Expert 2		Expert 3	
	Costs (C)	Costs (C)	APE (%)	Costs (C)	APE (%)	Costs (C)	APE (%)
Rovo3	516.76	545.08	5.48	554.19	7.24	561.83	8.72
Husky	331.76	347.93	4.87	365.64	10.21	349.74	5.42
Spot	571.63	574.44	0.49	581.39	1.71	575.02	0.59
Mercator	689.01	716.81	4.03	761.61	10.54	741.24	7.58

analysis. From there, it can be seen that the experts had other expectations for the planned path from Rovo3. On the other hand, the traversability estimates of Spot and the other robots seem to closely match the expectations of the experts. Furthermore, the costs of the predicted and experts' path were computed based on the derived cost table and environment information. Table 9 shows the computed costs values for the given paths and also provides an absolute percentage error (APE) to show how far the cost of the experts' paths deviate from the prediction. The used cost maps reflect the traversability estimates from the recorded locomotion data. Investigating the APE between an expert's path and the prediction is a good indicator on how well the derived cost maps fit to the experts' estimations. It can be seen that while the paths from the experts often diverged from the computed path, the costs of their paths were still similar to the computed path.

5. Discussion

In this section, we discuss the results obtained from the conducted experiments in more detail. The proposed pipeline was applied to recorded data of four different robot systems to evaluate its ability to generate robot-dependent off-road cost maps (semi)automatically. The resulting cost maps in Fig. 6 show, that the traversability of the terrain highly depends on the locomotion capabilities of the used robot. While Husky and Rovo3 are able to traverse most of the environments, Mercator and Spot struggle, especially with steep terrain and high vegetation. This also reflects the authors' observations while recording the data on the testing ground. In the resulting cost tables (e.g. see Table 6), noise and imprecisions could be observed for some classes. For example, the cost of land cover "MV" and slope "0" does not fully reflect the actual capability of the system as it seems that costs get worse for easier terrain. Similarly to that, land cover class "SL" and slope class "−3" has lower cost, although the terrain is steeper. Although going downwards seems to reduce the costs in some classes, this seems to be an outlier, as the lower class (SL, −2) is marked to be completely non-traversable.

For a quantitative evaluation of the cost estimation, path segments from the recorded locomotion data were excluded from the cost estimation process and used to compare the estimated costs to the actual costs. The overall prediction accuracy of 20–25% (see Table 7) on all four robots shows that a statistical extraction of a cost estimate is useful for a rough estimate of the traversability for given terrain classes. However, there is a higher variance within the robots that were evaluated. While some perform badly (e.g. Husky on DR, negative slope with an

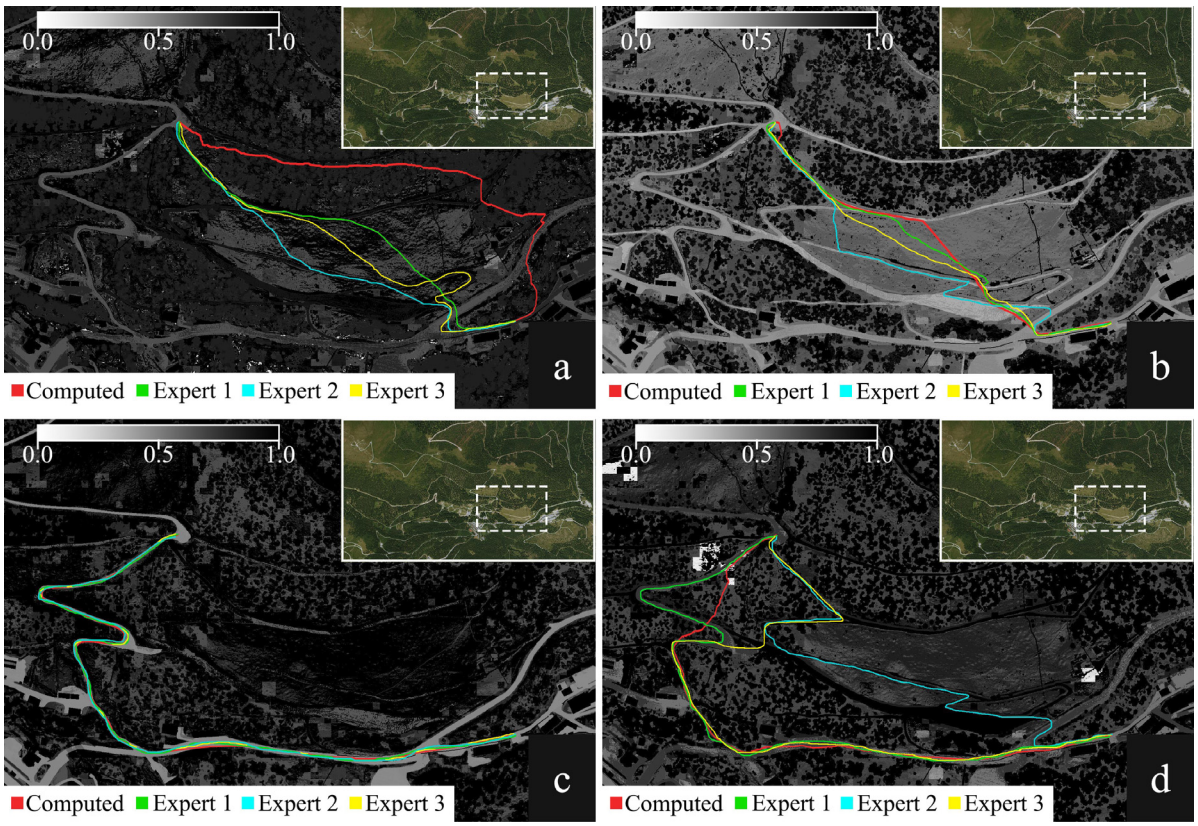

Fig. 10. Path planning on the computed cost map using A* for (a) Rovo3, (b) Husky, (c) Spot, (d) and Mercator. Costs are shown in grayscale. The red path indicates the planned path through the terrain using the computed cost estimates. Green, turquoise, and yellow show the expectations of the experts. For Spot (c), all paths overlap strongly due to their similarity. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

error of 132.29%), others show only a slight difference in the results (e.g. Rovo3 on AM, RO, positive slope with an error of 0.23%). When comparing the path segments with the same terrain classes on all four robots (Table 7 right), two classes seem to fall out of the average error. "DR, negative slope" is with 55.31% way above the average error. Possible reasons for this could be the lack of recorded data points for certain terrain classes or the high variance of the extracted features in a terrain class that cannot be represented by a single aggregated feature value. To overcome this, the proposed pipeline could be adapted to estimate the traversability of a given terrain by also taking information about the robot's immediate surroundings into account. On the other hand, "RO, positive slope" has a much lower error of 7.94%, which could be explained by a low variance within the terrain class, as roads tend to provide a smooth and uniform driving behavior.

Since robots often differ greatly in various aspects (e.g. dimensions, weight, locomotion...), it is difficult to draw general conclusions for the traversability of a robot type on given terrain. For example, it is not clear which capability or specification of the robot influences the calculation and in which way. To derive general conclusions, the experiments would have to be repeated with several platforms that differ only slightly from each other.

Path planning on the computed cost maps results in different routes suggested for all four robots. A direct comparison from the expert interviews shows, that the paths reflect the actual capabilities of the robotic systems quite well. However, it has to be noted that the subjective assessment of the optimal routes for a given robot system by an expert is dependent on his/her subjective awareness of risk. No qualitative evaluations of the paths are performed so far and thus need to be conducted in the future.

A downside of the proposed method is, that the recorded dataset depends on the operator's handling of the robot. As the robot is manually controlled by the operator, the dataset can be biased by the operator's control skills, even though the operator is instructed to follow a uniform diving behavior. Therefore, if the method is used to compare the capabilities of different robots within a specific environment, it is recommended to have the same operator handling all robotic systems. Alternatively, assistance modules or shared autonomy components could be used to mitigate this problem during data recording (e.g. [49,50]).

6. Conclusion

This paper presented a new pipeline for traversability analysis using locomotion data recorded in standardized experiments and segmented earth observation data. By aggregating locomotion features obtained from the recorded data which represents the robot's performance over certain terrain, the traversability of a given terrain can be estimated using a weighted cost function. Costs for terrain classes that were not sufficiently represented in the recorded data can be interpolated, if enough costs from ordinal neighbor classes exist. Experiments on the Seetaler Alps, an alpine terrain in Austria, were conducted using four robot systems with different locomotion systems (tracked, legged, differential drive, double Ackermann drive). While the experiments show that the results depend on the quality and amount of recorded data, the computed traversability costs well reflect the capabilities of the robotic systems. It is also shown, that assigning a general cost estimate for a given terrain class is representative if the locomotion behavior in the terrain class itself is stable and does not show a high variance. For terrain classes with high variance, additional information (e.g. on the immediate surrounding of the robot) should be integrated to achieve a more fine-grained traversability estimate. The recorded datasets are publicly

available and can be used for further research on traversability analysis or similar topics. Future work focuses on the optimization of the proposed pipeline with the help of machine learning approaches to improve the traversability estimation on terrain which is known to vary even within its class. Another research direction is how the results of the existing cost estimation can be transferred to other robot systems without the need to repeat the full set of experiments. Here, general descriptive parameters of robot properties and capabilities in combination with transfer learning could be investigated.

CRediT authorship contribution statement

Matthias Eder: Conceptualization, Investigation, Methodology, Experimentation, Evaluation, Visualization, Writing – original draft. **Raphael Prinz:** Experimentation, Evaluation, Data curation. **Florian Schöggl:** Conceptualization, Evaluation. **Gerald Steinbauer-Wagner:** Conceptualization, Experimentation, Writing – review & editing, Supervision, Project Lead.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The link to the data used is shared in the paper.

Acknowledgment

This work was partially funded by the Austrian Research Promotion Agency (FFG) with the project RoboNav.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.robot.2023.104494.

References

[1] P. Wolf, K. Berns, Data-fusion for robust off-road perception considering data quality of uncertain sensors, in: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, 2021, pp. 6876–6883, http://dx.doi.org/10.1109/IROS51168.2021.9636541.

[2] B. Gao, S. Hu, X. Zhao, H. Zhao, Fine-grained off-road semantic segmentation and mapping via contrastive learning, in: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, 2021, pp. 5950–5957.

[3] Z. Zhu, N. Li, R. Sun, D. Xu, H. Zhao, Off-road autonomous vehicles traversability analysis and trajectory planning based on deep inverse reinforcement learning, in: 2020 IEEE Intelligent Vehicles Symposium, IV, 2020, pp. 971–977, http://dx.doi.org/10.1109/IV47402.2020.9304721.

[4] M. Hamandi, D. Asmar, E. Shammas, Ground segmentation and free space estimation in off-road terrain, Pattern Recognit. Lett. 108 (2018) 1–7, http://dx.doi.org/10.1016/j.patrec.2018.02.019.

[5] S. Sharma, J.E. Ball, B. Tang, D.W. Carruth, M. Doude, M.A. Islam, Semantic segmentation with transfer learning for off-road autonomous driving, Sensors (Basel, Switzerland) 19 (2019).

[6] T. Cortinhal, G. Tzelepis, E. Erdal Aksoy, SalsaNext: Fast, uncertainty-aware semantic segmentation of LiDAR point clouds, in: Advances in Visual Computing: 15th International Symposium, ISVC 2020, San Diego, CA, USA, October 5–7, 2020, Proceedings, Part II, Springer-Verlag, Berlin, Heidelberg, 2020, pp. 207–222, http://dx.doi.org/10.1007/978-3-030-64559-5_16.

[7] S.M. Haider Jafri, R. Kala, Path planning of a mobile robot in outdoor terrain, in: S. Berretti, S.M. Thampi, S. Dasgupta (Eds.), Intelligent Systems Technologies and Applications, Springer International Publishing, Cham, 2016, pp. 187–195.

[8] R. Prinz, R. Bulbul, J. Scholz, M. Eder, G. Steinbauer-Wagner, Off-road navigation maps for robotic platforms using convolutional neural networks, AGILE: GIScience Ser. 3 (2022) 55, http://dx.doi.org/10.5194/agile-giss-3-55-2022, URL https://agile-giss.copernicus.org/articles/3/55/2022/.

[9] J. Collier, G. Broten, J. Giesbrecht, in: Defense Technical Information Center (Ed.), Traversability Analysis for Unmanned Ground Vehicles: Interpreting the Environment, Defence Research and Development Suffield, 2006.

[10] A. Jacoff, E. Messina, J. Evans, Performance evaluation of autonomous mobile robots, Ind. Robot Int. J. 29 (3) (2002) 259–267, http://dx.doi.org/10.1108/01439910210425568.

[11] M. Wei, V. Isler, Predicting energy consumption of ground robots on uneven terrains, IEEE Robot. Autom. Lett. 7 (1) (2022) 594–601, http://dx.doi.org/10.1109/LRA.2021.3130630.

[12] M. Quann, L. Ojeda, W. Smith, D. Rizzo, M. Castanier, K. Barton, Off–road ground robot path energy cost prediction through probabilistic spatial mapping, J. Field Robotics 37 (3) (2020) 421–439, http://dx.doi.org/10.1002/rob.21927.

[13] M. Eder, R. Prinz, F. Schoggl, G. Steinbauer-Wagner, Generating robot-dependent cost maps for off-road environments using locomotion experiments and earth observation data, in: 2022 Sixth IEEE International Conference on Robotic Computing, IRC, IEEE, 2022, pp. 172–176, http://dx.doi.org/10.1109/IRC55401.2022.00036.

[14] A.L. Rankin, A. Huertas, L.H. Matthies, Stereo-vision-based terrain mapping for off-road autonomous navigation, in: G.R. Gerhart, D.W. Gage, C.M. Shoemaker (Eds.), Unmanned Systems Technology XI, in: SPIE Proceedings, SPIE, 2009, 733210, http://dx.doi.org/10.1117/12.819099.

[15] H. Roncancio, M. Becker, A. Broggi, S. Cattani, Traversability analysis using terrain mapping and online-trained terrain type classifier, in: Intelligent Vehicles Symposium Proceedings, 2014 IEEE, IEEE, 2014, pp. 1239–1244, http://dx.doi.org/10.1109/IVS.2014.6856427.

[16] W. Walch, M. Eder, K. Mautner-Lassnig, G. Steinbauer-Wagner, Offroad terrain classification for mobile robots, in: Proceedings of the ARW 2022, 2022, pp. 6–11.

[17] G.G. Waibel, T. Löw, M. Nass, D. Howard, T. Bandyopadhyay, P.V.K. Borges, How rough is the path? Terrain traversability estimation for local and global path planning, IEEE Trans. Intell. Transp. Syst. (2022) 1–12, http://dx.doi.org/10.1109/TITS.2022.3150328.

[18] A. Broggi, E. Cardarelli, S. Cattani, M. Sabbatelli, Terrain mapping for off-road autonomous ground vehicles using rational b-spline surfaces and stereo vision, in: Intelligent Vehicles Symposium (IV), 2013 IEEE, IEEE, 2013, pp. 648–653, http://dx.doi.org/10.1109/IVS.2013.6629540.

[19] V. Khryashchev, L. Ivanovsky, V. Pavlov, A. Ostrovskaya, A. Rubtsov, Comparison of different convolutional neural network architectures for satellite image segmentation, in: S.I. Balandin (Ed.), Proceedings of the 23rd Conference of Open Innovations Association FRUCT, IEEE, Piscataway, NJ, 2018, pp. 172–179, http://dx.doi.org/10.23919/FRUCT.2018.8588071.

[20] H. Didari, M. Eder, G. Grömer, R. Halatschek, S. Özdemir-Fritz, R. Prinz, J. Scholz, G. Steinbauer-Wagner, The AMADEE-20 robotic exploration cascade: an experience report, in: A. Müller, M. Brandstötter (Eds.), Advances in Service and Industrial Robotics, Springer International Publishing, Cham, 2022, pp. 477–484, http://dx.doi.org/10.1007/978-3-031-04870-8_56.

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

[22] B. Russell, A. Torralba, K. Murphy, W. Freeman, LabelMe: A database and web-based tool for image annotation, Int. J. Comput. Vis. 77 (1–3) (2008) 157–173, http://dx.doi.org/10.1007/s11263-007-0090-8.

[23] D. Maturana, P.-W. Chou, M. Uenoyama, S. Scherer, Real-time semantic mapping for autonomous off-road navigation, in: Field and Service Robotics, Springer, 2018, pp. 335–350.

[24] H. Abu Alhaija, S.K. Mustikovela, L. Mescheder, A. Geiger, C. Rother, Augmented reality meets computer vision: Efficient data generation for urban driving scenes, Int. J. Comput. Vis. 126 (9) (2018) 961–972, http://dx.doi.org/10.1007/s11263-018-1070-x.

[25] P. Jiang, P.R. Osteen, M. Wigness, S. Saripalli, RELLIS-3D dataset: Data, benchmarks and analysis, CoRR abs/2011.12954, 2020, URL https://arxiv.org/abs/2011.12954.

[26] B. Douillard, J. Underwood, N. Kuntz, V. Vlaskine, A. Quadros, P. Morton, A. Frenkel, On the segmentation of 3D LIDAR point clouds, in: 2011 IEEE International Conference on Robotics and Automation, 2011, pp. 2798–2805, http://dx.doi.org/10.1109/ICRA.2011.5979818.

[27] F.G. Oliveira, E.R.S. Santos, A.A. Neto, M.F.M. Campos, D.G. Macharet, Speed-invariant terrain roughness classification and control based on inertial sensors, in: 2017 Latin American Robotics Symposium, LARS, IEEE, 2017, pp. 1–6, http://dx.doi.org/10.1109/SBR-LARS-R.2017.8215332.

[28] P. Papadakis, Terrain traversability analysis methods for unmanned ground vehicles: A survey, Eng. Appl. Artif. Intell. 26 (4) (2013) 1373–1385, http://dx.doi.org/10.1016/j.engappai.2013.01.006.

[29] Z. Jian, Z. Lu, X. Zhou, B. Lan, A. Xiao, X. Wang, B. Liang, PUTN: A plane-fitting based uneven terrain navigation framework, 2022, http://dx.doi.org/10.48550/ARXIV.2203.04541, URL https://arxiv.org/abs/2203.04541.

[30] C. Sevastopoulos, S. Konstantopoulos, A survey of traversability estimation for mobile robots, IEEE Access 10 (2022) 96331–96347, http://dx.doi.org/10.1109/ACCESS.2022.3202545.
[31] P. Čížek, M. Zoula, J. Faigl, Design, construction, and rough-terrain locomotion control of novel hexapod walking robot with four degrees of freedom per leg, IEEE Access 9 (2021) 17866–17881, http://dx.doi.org/10.1109/ACCESS.2021.3053492.
[32] M. Prágr, P. Čížek, J. Faigl, Traversal cost modeling based on motion characterization for multi-legged walking robots, in: 2019 European Conference on Mobile Robots, ECMR, 2019, pp. 1–6, http://dx.doi.org/10.1109/ECMR.2019.8870912.
[33] M. Prágr, R. Szadkowski, J. Bayer, J. Zelinka, J. Faigl, Terrain traversal cost learning with knowledge transfer between multi-legged walking robot gaits, in: 2022 IEEE International Conference on Autonomous Robot Systems and Competitions, ICARSC, 2022, pp. 148–153, http://dx.doi.org/10.1109/ICARSC55462.2022.9784790.
[34] A. Reddy, M. Ordway-West, M. Lee, M. Dugan, J. Whitney, R. Kahana, B. Ford, J. Muedsam, A. Henslee, M. Rao, Using Gaussian mixture models to detect outliers in seasonal univariate network traffic, in: 2017 IEEE Security and Privacy Workshops, SPW, 2017, pp. 229–234, http://dx.doi.org/10.1109/SPW.2017.9.
[35] W. Liu, D. Cui, Z. Peng, J. Zhong, Outlier detection algorithm based on Gaussian mixture model, in: 2019 IEEE International Conference on Power, Intelligent Computing and Systems, ICPICS, 2019, pp. 488–492, http://dx.doi.org/10.1109/ICPICS47731.2019.8942474.
[36] K.M. Clark, in: MacSphere (Ed.), Outlier Detection in Gaussian Mixture Models, McMaster Univeristy, 2020, URL http://hdl.handle.net/11375/25930.
[37] R. Domingues, M. Filippone, P. Michiardi, J. Zouaoui, A comparative evaluation of outlier detection algorithms: Experiments and analyses, Pattern Recognit. 74 (2018) 406–421, http://dx.doi.org/10.1016/j.patcog.2017.09.037.
[38] A.V. Oppenheim, R.W. Schafer, Discrete-Time Signal Processing, third ed., pearson new international ed., in: Always Learning, Pearson, Harlow, 2014, URL https://ebookcentral.proquest.com/lib/kxp/detail.action?docID=5832333.
[39] R. Dechter, Constraint Processing, in: The Morgan Kaufmann Series in Artificial Intelligence, Elsevier, San Francisco, CA, USA, 2003, http://dx.doi.org/10.1016/B978-1-55860-890-0.X5000-2.
[40] S. Greco, R. Słowiński, J.R. Figueira, V. Mousseau, Robust ordinal regression, in: M. Ehrgott, J.R. Figueira, S. Greco (Eds.), Trends in Multiple Criteria Decision Analysis, in: International Series in Operations Research & Management Science, vol. 142, Springer US, Boston, MA, 2010, pp. 241–283, http://dx.doi.org/10.1007/978-1-4419-5904-1_9.
[41] R. Halatschek, K. Ramanna, W. Url, G. Steinbauer-Wagner, Universal offroad robot platform for disaster response, in: 2020 IEEE International Symposium on Safety, Security, and Rescue Robotics, SSRR, 2020, p. 6.
[42] S. Balasubramanian, A. Melendez-Calderon, E. Burdet, A robust and sensitive metric for quantifying movement smoothness, IEEE Trans. Biomed. Eng. 59 (8) (2012) 2126–2136, http://dx.doi.org/10.1109/TBME.2011.2179545.
[43] W. Wen, Road roughness detection by analyzing IMU data, in: TRITA-GIT EX, School of Architecture and the Built Environment, Royal Institute of Technology (KTH), 2008, URL http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-199681.
[44] J. Han, J. Pei, M. Kamber, Data Mining: Concepts and Techniques, in: The Morgan Kaufmann Series in Data Management Systems, Elsevier Science, 2011, http://dx.doi.org/10.1016/B978-0-12-381479-1.00001-0, URL https://www.sciencedirect.com/science/article/pii/B9780123814791000010.
[45] P.E. Hart, N.J. Nilsson, B. Raphael, A formal basis for the heuristic determination of minimum cost paths, IEEE Trans. Syst. Sci. Cybern. 4 (2) (1968) 100–107, http://dx.doi.org/10.1109/TSSC.1968.300136.
[46] D.V. Lu, D. Hershberger, W.D. Smart, Layered costmaps for context-sensitive navigation, in: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2014, pp. 709–715, http://dx.doi.org/10.1109/IROS.2014.6942636.
[47] S.L. Seyler, A. Kumar, M.F. Thorpe, O. Beckstein, Path similarity analysis: A method for quantifying macromolecular pathways, PLoS Comput. Biol. 11 (10) (2015) e1004568, http://dx.doi.org/10.1371/journal.pcbi.1004568.
[48] H. Alt, M. Godau, Computing the Fréchet distance between two polygonal curves, Int. J. Comput. Geom. Appl. 05 (01n02) (1995) 75–91, http://dx.doi.org/10.1142/S0218195995000064.
[49] Y. Okada, K. Nagatani, K. Yoshida, T. Yoshida, E. Koyanagi, Shared autonomy system for tracked vehicles to traverse rough terrain based on continuous three-dimensional terrain scanning, in: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2010, pp. 357–362, http://dx.doi.org/10.1109/IROS.2010.5648974.
[50] M. Gao, R. Kohlhaas, J.M. Zöllner, Contextual learning and sharing autonomy to assist mobile robot by trajectory prediction, in: 2016 IEEE International Symposium on Safety, Security, and Rescue Robotics, SSRR, 2016, pp. 274–275, http://dx.doi.org/10.1109/SSRR.2016.7784312.

Matthias Eder received his B.Sc. and M.Sc. degree in Software Engineering and management from Graz University of Technology in 2016 and 2017. Since October 2020, he is a Ph.D. candidate at the Institute of Software Technology, Graz University of Technology. His research interests lie in off-road navigation, explainable artificial intelligence, and human–robot interaction. He successfully participated in multiple robotic competitions.

Raphael Prinz received his B.Sc in Geomatics Engineering in 2019 from Graz University of Technology. Currently, he is a graduate student at the Institute of Geodesy, Graz University of Technology where he is also doing his master thesis. In the master thesis, he focuses on remote sensing and the use of earth observation data for robot navigation. His research interests include robotics, remote sensing, and geo informatics.

Florian Schöggl received his B.Sc. in Geomatics Engineering in 2008 and his M.Sc in Geomatics Science in 2011 from Graz University of Technology. Afterwards, he worked as a Scientific Project Assistant at the Institutes of Geoinformation, Business Informatics and Remote Sensing and Photogrammetry at Graz University of Technology. Currently, he is CEO of pentamap GmbH and involved in several research projects. His research interests include to design and develop smart geospatial software solutions for mobile devices, and grid- and vector-based navigation approaches for humans and different vehicle types in urban as well as remote areas.

Gerald Steinbauer-Wagner received an M.Sc. in Computer Engineering and a Ph.D. in Computer Science in 2001 and 2006 from Graz University of Technology. He is currently associate professor at the Institute of Software Technology at the Graz University of Technology and leads the working group for Autonomous Intelligent Systems. His research interests include robot control, cognitive robotics, knowledge representation, reasoning, and model-based diagnosis. He is a member of the IEEE Robotics and Automation Society, ACM and the Austrian Society for Artificial Intelligence. Moreover, he is founding president of the Austrian RoboCup National Chapter and the Austrian IEEE Robotics and Automation Chapter.