Mechatronics 62 (2019) 102260

Contents lists available at ScienceDirect

Mechatronics

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

Support vector machine based real-time terrain estimation for tracked robots ☆

Bijo Sebastian, Pinhas Ben-Tzvi*

Department of Mechanical Engineering, Virginia Tech, Blacksburg, VA 24060, USA

ARTICLE INFO

Keywords:
Mobile robot
Estimation
Classification
Real-time systems
Navigation

ABSTRACT

This paper describes the use of support vector machine (SVM) classifier for real-time terrain estimation to improve autonomous navigation of tracked mobile robots. Real-time terrain identification and terrain property estimation has been explored previously for application on a wide range of systems from planetary exploration rovers to commercial vehicles like cars and trucks. A majority of the existing methods rely on the use of dedicated sensors including vibration sensors, accelerometers, cameras, LIDAR, etc., which makes them susceptible to the failure modes of each of these sensors. This work proposes a method for real-time classification of different terrain types based on the state evolution of a ground robot, specifically the measured change in the pose of the robot for a known control input. By using a trained SVM to perform terrain estimation based on the collected state evolution data, the proposed method does not require dedicated sensing modalities solely for the terrain estimation. In addition, this method is generally applicable in all conditions where the robot can traverse. The training data was obtained from four different terrain conditions including vinyl flooring, asphalt, artificial turf, and grass–gravel, to train the SVM to perform terrain estimation. The proposed technique is validated using a skid-steer tracked robot over multiple simulated and real terrain transitions cases, where the response to control inputs is significantly affected by terrain characteristics. The results show that the proposed method provides greater than 80% accuracy in all cases, with fast detection of terrain transitions. The paper concludes with a detailed description on the application of real-time terrain estimation in improving autonomous navigation.

1. Introduction

Autonomous navigation of mobile robots has advanced greatly in recent years; this is evident in the world wide development of self-driving technology, advanced driver assistance features that are commonplace in most commercial vehicles, and the wide spread commercial use of warehouse automation robots. Despite these breakthrough achievements, autonomous navigation in its true sense, especially in unstructured terrain conditions, is still an unsolved problem. This is evident from the fact that the majority of ground vehicles deployed in search and rescue operations in the past decades were remotely operated [1–3] and that recent efforts by the military toward the use of Squad Multipurpose Equipment Transport (SMET) [4,5] systems have also been focused toward semi-autonomous capabilities, including leader following approaches and remote guidance.

While navigating in unstructured terrain conditions, through autonomous or tele-operated manner, performing high level planning and control requires detailed knowledge of the terrain. Vision based methods can be used for terrain estimation, but they are highly susceptible to changes in ambient light and other environment conditions such as presence of smoke, fog, or dust. Other sensor based approaches including LIDAR and Inertial Measurement Units (IMU) are also not robust enough. For example, while a rocky slope and an expanse of loose sand at same gradient may look similar in a 3D scan, the robot can fail to climb or become entrenched on the sandy slope depending on a number of factors. Currently, for systems that are deployed in the field, high level navigation decisions are made by a human operator based on feedback from the camera or LIDAR systems. The most recent DARPA Subterranean (SubT) Challenge [6] requires roboticists to address the problem of terrain estimation in order to enable fully autonomous navigation in unstructured conditions.

In addition to improving the autonomous navigation capabilities of terrestrial rovers, and search and rescue robots, real-time terrain estimation can also benefit commercial vehicles such as cars and trucks through improvements in advanced driver assistance systems (ADAS) which in turn leads to fully autonomous driving. Generating accurate predictions about how the vehicle is going to respond to various maneuvering commands from the onboard planning system is vital for safe

☆ This paper was recommended for publication by Associate Editor D. Hoelzle.

* Corresponding author.
E-mail address: bentzvi@vt.edu (P. Ben-Tzvi).

https://doi.org/10.1016/j.mechatronics.2019.102260
Received 16 January 2019; Received in revised form 25 July 2019; Accepted 9 August 2019
0957-4158/

autonomous driving. In this regard, ascertaining road terrain parameters is crucial, as vehicle behavior and the safe limits, including maximum safe acceleration and braking limits, vary drastically with changes in the terrain. There exist studies by different groups in using a wide variety of sensing modalities mounted on vehicles including radar [7], images from a monocular camera [8], and RGBD point clouds obtained from a stereo camera pair [9] to estimate the terrain conditions. In most cases, the goal was long range binary classification of the terrain into navigable and non-navigable areas aimed at both on-road and off-road autonomous navigation.

In comparison to the existing research, this work proposes a method for real-time classification of different terrain types based on the state evolution of a ground robot, specifically the measured change in pose of the robot for a known control input. An offline technique to determine the features of interest from the collected training data is presented. A support vector machine (SVM) classifier is trained on the principal components (PCs) of the weights associated with the features of interest to perform terrain estimation. The online moving window least squares estimation technique used to update the weights in real-time is also presented along with the SVM architecture.

By using a trained SVM to perform terrain estimation based on the collected state evolution data, the proposed method does not require dedicated sensing modalities solely for terrain estimation. As such, this method is generally applicable in all conditions where the robot can traverse. The training data was obtained from four different terrain conditions including vinyl flooring, asphalt, artificial turf, and grass–gravel. The proposed technique is validated using a skid-steer tracked robot, where the response to control inputs is significantly affected by terrain characteristics, over multiple simulated and real terrain transitions cases.

It should be noted that the proposed technique enables the robot to identify the terrain it is operating on, based on previously learned terrain types. This requires the robot to travel over a given terrain condition and collect data regarding how the robot pose changes for a given control input. As such it cannot be used to classify terrain into navigable versus non-navigable as the robot cannot collect motion data on non-navigable terrain. Readers interested in navigable versus non-navigable classification of terrain should refer to prior work by the authors that specifically addressed this issue [10].

The rest of the paper is organized as follows: Section 2 provides a detailed literature review in this domain including prior work on terrain estimation for planetary rovers and commercial vehicles. Section 3 describes the shortcomings of the existing methods, outlines the proposed method along with its novelty, and explains how it addresses the shortcomings of existing techniques. Section 4 presents experimental validation of the proposed technique, Section 5 describes techniques to improve autonomous navigation using the terrain estimation results, and Section 6 concludes the work with directions for future research.

2. Literature review

Planetary exploration missions are a major application area where online terrain parameter estimation is used to improve navigation. Terrain estimation allows the rovers to operate with minimal human supervision in terrain conditions that are often challenging and previously unseen. To this extent, Iagnemma et al. [11] used a simplified form of classical terramechanics equations along with a linear-least squares estimator to compute terrain parameters in real-time. Since their approach relied on terramechanics equations, it assumed that the vertical load on the robot, torque applied on motors, sinkage of the robot, wheel angular speed, and wheel linear speed could be measured or estimated. This requires additional sensors on the robot. On a similar note, Reina et al. [12] proposed methods for wheel slippage and sinkage detection as applied to planetary rovers by comparing wheel encoder readings with gyro readings and motor current in order to detect slip. They also proposed a novel vision based method to detect wheel sinkage.

One of the major disadvantages of vision based methods is that they are susceptible to variations in ambient light and the presence of dust, smoke, or other environmental conditions. In addition, vision and range based methods are not good at detecting non-geometric hazards, such as in cases when the topmost layer of terrain is different from the underlying load bearing surface. To handle these issues Iagnemma and Brooks [13] have explored terrain classification based on the vibrations induced on the rover by the wheel-terrain interaction during driving. Their approach aimed at creating a stand-alone classifier by performing standard signal processing techniques on the data collected from a vibration sensor mounted on the wheel of the rover. On a similar note, Giguere and Dudek [14] attached an accelerometer at the end of a metallic rod to form a tactile probe. The acceleration patterns induced at the tip of the rod were analyzed to estimate the terrain conditions using a trained neural network. Similarly, Park et al. [15] used peak variances extracted from contact sensor data to perform terrain classification. Wolf and Sukhatme proposed the use of Hidden Markov Models (HMMs) and SVMs for semantic terrain mapping [16]. They used data provided by range sensors and odometers to classify the terrain into navigable and non-navigable regions. Weiss et al. [17] proposed the use of an SVM trained over raw vibration sensor data collected by the robot to perform terrain classification. They proposed that based on the terrain estimation data, the vehicle could adapt its driving style to better match the terrain being traversed. Their method was compared to five other existing vibration-based terrain classification methods in [18] and the results showed that significantly better performance. An interesting example of the application of environment identification to improve autonomous navigation was presented by Giguere et al. in [19]. They performed environment identification based on actuator and inertial sensor data to autonomously switch between walking and swimming for their amphibious hexapod, AQUA.

In addition to rovers, real-time estimation of terrain parameters has been explored in the context of commercial vehicles including cars and trucks. As mentioned by Wang et al. in [20], real-time terrain estimation can improve the performance of driver assistance systems by specifying terrain specific driving strategies aimed at improving factors like fuel efficiency. They proposed a two-stage road terrain identification approach for land vehicles using feature-based and Markov random field algorithms. As per their approach, feature-based identification results obtained using an accelerometer, camera, and LRF was improved upon by using a Markov Random Field (MRF) to get optimal identification results. A more recent work by Khaleghian and Taheri looks at terrain classification methods using an intelligent tire [21]. The intelligent tire setup essentially consists of a tri-axial accelerometer attached to the tire inner liner, wheel speed sensors, and an additional accelerometer on the vehicle chassis along with a data acquisition system. In addition, the robot also had free wheels for accurately measuring slip. Based on the accelerometer readings and the wheel slip ratio, a fuzzy logic algorithm was used to perform terrain classification.

The survey by Khaleghian et al. [22], lists recent advances in tire-road friction estimation toward improving vehicle's stability, traction, and ABS controller performance for cars. This work provides a detailed summary of existing approaches including optical, acoustic, and tire tread sensors in addition to model based methods toward estimating tire-road friction. Even though commercial vehicles could also benefit greatly from terrain estimation and road parameter estimation, the rest of the paper will focus on these techniques as applied to the motion of tracked skid steer robots in varying terrain conditions. The applicability to commercial vehicles will be further addressed as part of future work.

3. Proposed real-time terrain estimation method

A main point to notice based on the above review is that almost all existing works use dedicated sensing systems solely for estimating the terrain type or to estimate the terrain characteristics like roughness, friction, etc. A majority of the terrain estimation methods use contact sensors, non-contact sensors, or a combination of both. As such these methods are susceptible to the failure modes on each sensing type. On the other hand this paper tries to explore the idea of using the robot itself as the sensor. This work is based on the hypothesis that the state evolution of the robot on any given terrain contains sufficient information to accurately perform terrain estimation, provided the control inputs applied to the robot are known. This eliminates the need for dedicated sensors, which removes the associated limitations and failure modes, while also making the method generally applicable in all possible terrain conditions, where the robot can navigate. A similar approach toward estimating wind parameters using an RC helicopter has been explored before [23]. In addition to the real-time terrain estimation method, this paper proposes potential techniques to improve navigation, especially in the areas of control, fault monitoring, autonomous path planning, and improving localization of mobile robotic systems traversing over varying terrain conditions.

The rest of this section is divided into three subsections, the first subsection explores the use of linear regression techniques to identify the features of interest that capture the effects of the terrain over which the robot is moving, the second subsection introduces methods for real-time estimation of the weights associated with the features of interest and the final subsection introduces principal component analysis (PCA) to reduce the dimensionality of the data, along with the support vector machine (SVM) classifier to perform terrain classification based on the PCs of the estimated weight values. The overall method is summarized at the end of this section in Fig. 5.

3.1. Identifying features of interest

Recently there has been significant interest toward using machine learning based model fitting to identify the features of interest that represent a physical process [24,25]. The overall approach can be summarized as follows: experimental data is used to create an over-complete library of possible features of interest, consisting of the variables involved in the process, nonlinear functions of these variables and even their products. A regression method is then used to find the best fitting model over the experimental data. Based on the best fitting model, features that show high weights can be easily identified as the features of interest. These would be the features that contribute the most toward the governing equations that represent the physical process. The above approach was used to identify the features of interest for the motion of a tracked skid steer robot over varying terrain conditions. Experimental data was obtained by driving the robot STORM [26] over four different terrain conditions with ten trails on each terrain. The terrain conditions consisted of asphalt, vinyl flooring, artificial turf, and a combination of grass and gravel. Each trial consisted of driving the robot in a square over the terrain with side length of 4–6 m.

For the data collection trials the robot was manually driven over regions where the terrain conditions were uniform within the POZYX sensing limits. The path followed by the robot for the duration of the trial was recorded using the absolute positioning system POZYX [27]. The control inputs were recorded using the wheel encoders on the robot. Each instance of the recorded data consists of previous 2D pose of the robot including position $(x_t,\ y_t)$ and yaw angle θ_t, control inputs given to the robot in terms of linear and angular velocity $(V_t,\ \omega_t)$, along with the current position and orientation of the robot $(x_{t+1},\ y_{t+1},\ \theta_{t+1})$. The experimental platform and the terrain over which it was run are shown in Fig. 1. Since the experimental data was collected with a light weight robot moving at low speeds, with negligible inertial effects, a kinematic analysis of the motion was considered sufficient.

The motion of differential drive platforms with one or more supporting castor wheels can be modeled accurately with a simple unicycle robot model, as given below.

$$\begin{aligned} x_{t+1} &= x_t + V_t\cos\theta_t\Delta t \\ y_{t+1} &= y_t + V_t\sin\theta_t\Delta t \\ \theta_{t+1} &= \theta_t + \omega_t\ \Delta t \end{aligned} \tag{1}$$

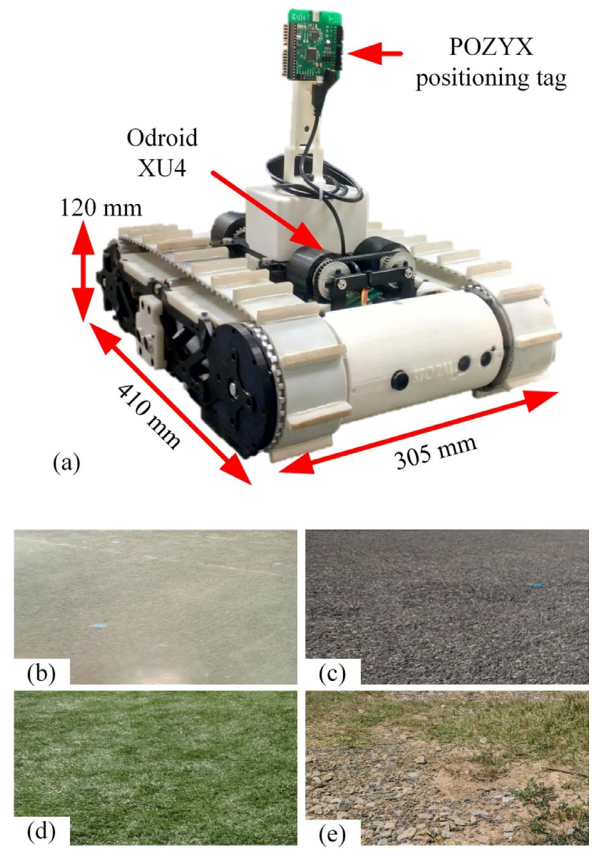

Fig. 1. The experimental setup and the various terrain conditions for data collection: (a) STORM, (b) vinyl flooring, (c) asphalt, (d) artificial turf, and (d) grass–gravel.

where $(x,\ y)$ is the 2D position robot fixed frame {R} with respect to the global inertial frame {G}, and θ represents the orientation of {R}, with respect to {G} about the Z-axis. In Eq. (1) Δt denotes the time gap between the current robot pose (x_t, y_t, θ_t) and the pose at the next time step $(x_{t+1},\ y_{t+1},\ \theta_{t+1})$. Existing studies [28,29] have shown that the motion of skid steer platforms, especially in varying terrain conditions, deviate considerably from the ideal model given by Eq. (1). Modifications to the ideal model have been suggested recently, including instantaneous center of rotation (ICR) based kinematics models [28,30], to better approximate the motion of skid steer robots on varying terrain conditions. In contrast to existing approaches, we perform linear regression on the collected data to determine the features of interest and thereby formulate a more accurate governing equation for the motion of tracked skid steer robots in a data driven manner. The linear regression model used on the data takes the form of:

$$\mathbf{Y} = \mathbf{wx} + \mathbf{b} \tag{2}$$

where the feature vector and the output vector are given by

$$\begin{aligned} \mathbf{x} &= \left[x_t, y_t, \theta_t, V_t\Delta t, \omega_t\Delta t, V_t\cos\theta_t\Delta t, V_t\sin\theta_t\Delta t, \omega_t\cos\theta_t\Delta t, \omega_t\sin\theta_t\Delta t\right]^T \\ \mathbf{Y} &= \left[x_{t+1}, y_{t+1}, \theta_{t+1}\right]^T \end{aligned} \tag{3}$$

For Eq. (2), **w** is the matrix of weights that linearly map the feature vector **x** to output vector **Y,** k is the total number of features in the over complete library, and **b** contains the constant bias term for each equation. For the chosen feature vector **x** shown in Eq. (3), the total number of features k is nine. Therefore **w** is a 3×9 matrix, with w_{ij} being each element where i varies from 1 to 3 corresponding to linear regression models for $x_{t+1},\ y_{t+1},\ \theta_{t+1}$ and j varies from 1 to k. The feature vector, **x** was chosen based on the existing research that suggested possible

Table 1
Experimental data used for model fitting.

Terrain condition		Number of data points	R squared test score
Simulated model	x_{t+1}	7885	0.999
	y_{t+1}		0.999
	θ_{t+1}		0.989
Asphalt	x_{t+1}	7596	0.999
	y_{t+1}		0.999
	θ_{t+1}		0.992
Grass-gravel	x_{t+1}	7059	0.982
	y_{t+1}		0.967
	θ_{t+1}		0.993
Artificial turf	x_{t+1}	6673	0.994
	y_{t+1}		0.998
	θ_{t+1}		0.997
Vinyl flooring	x_{t+1}	6000	0.999
	y_{t+1}		0.998
	θ_{t+1}		0.996

modifications to the unicycle model to better approximate the motion of skid steer vehicle on varying terrain [29]. To verify the feasibility of the proposed method, the ideal unicycle model, as given by Eq. (1), was simulated in MATLAB, without any added noise. The state variables (x, y, θ) were initialized at zero and control inputs (V, ω) were applied to the model to make the robot move in a circle, straight line, sinusoidal path, and also turn in place. The linear regression method, applied to the experimental data, was applied to the simulated ideal data as well.

Performing least square regression on the experimental data allows us to determine the weights associated with each feature given in Eq. (3) for the governing equations corresponding to x_{t+1}, y_{t+1}, and θ_{t+1}. The number of data points collected for each terrain condition and the R squared test score for the best fit model are given in Table 1. As per the R squared test score, the models given by the regression algorithm provide a good fit on the experimental data. Fig. 2 shows the absolute value of the weights corresponding to each feature as obtained from the best fit model, for the different data sets. In Fig. 2, the features that show higher weights are the ones that contribute most toward the motion of the robot. By identifying the features that have high weights (>0.5) in at least one terrain condition, a more accurate governing equation representing the evolution of the robot's states can be formed:

$$\begin{aligned} x_{t+1} &= w_{11}x_t + w_{12}V_t\Delta t + w_{13}V_t\cos\theta_t\Delta t + w_{14}V_t\sin\theta_t\Delta t \\ y_{t+1} &= w_{21}y_t + w_{22}V_t\Delta t + w_{23}V_t\cos\theta_t\Delta t + w_{24}V_t\sin\theta_t\Delta t \\ \theta_{t+1} &= w_{31}\theta_t + w_{32}V_t\Delta t + w_{33}V_t\cos\theta_t\Delta t + w_{34}V_t\sin\theta_t\Delta t \\ &\quad + w_{35}\omega_t\Delta t + w_{36}\omega_t\cos\theta_t\Delta t + w_{37}\omega_t\sin\theta_t\Delta t \end{aligned} \tag{4}$$

The advantage of the above approach is that it allows us to pick out the features that are most relevant to the physical phenomenon, while ignoring all of the non-relevant ones, completely based on the collected experimental data. It should be noted that limiting the loop back condition on θ between $[-\pi, \pi)$, will result in the linear regression failing to converge due to the inherent non linearity. To counter this, the collected data was adjusted to remove the loop back of θ, allowing it to continuously increase or decrease without limits.

In addition to deciding upon the features of interest Fig. 2 also offers some additional insight worth discussing. The ideal weights for each feature as per Eq. (1) are given in blue. The weights predicted by least square fit for the simulated data is shown in magenta. As seen from the figure the ideal weights and the simulated data match perfectly as expected. This validates the fact that the linear regression fitting works as expected. On the other hand, the experimental data collected from each terrain gives a different set of weights as compared to the ideal model. For example, as per the ideal model x_{t+1} should depend only on x_t and $V_t\cos\theta_t\Delta t$, with weights equivalent to 1.0. This is true for the data generated from the ideal model, but the experimental data obtained from most of the terrain conditions depend on $V_t\sin\theta_t\Delta t$ as well. Similarly, the equations for y_{t+1} and θ_{t+1} deviate from the ideal model for the experimental data. Another important factor to note is that each terrain shows different optimal value of weights for a given feature of interest. It can be inferred that the variations in the optimal value of weights is characteristic to the terrain over which the robot is moving. In other words, the weights associated with the features of interest holds characteristic information about the terrain.

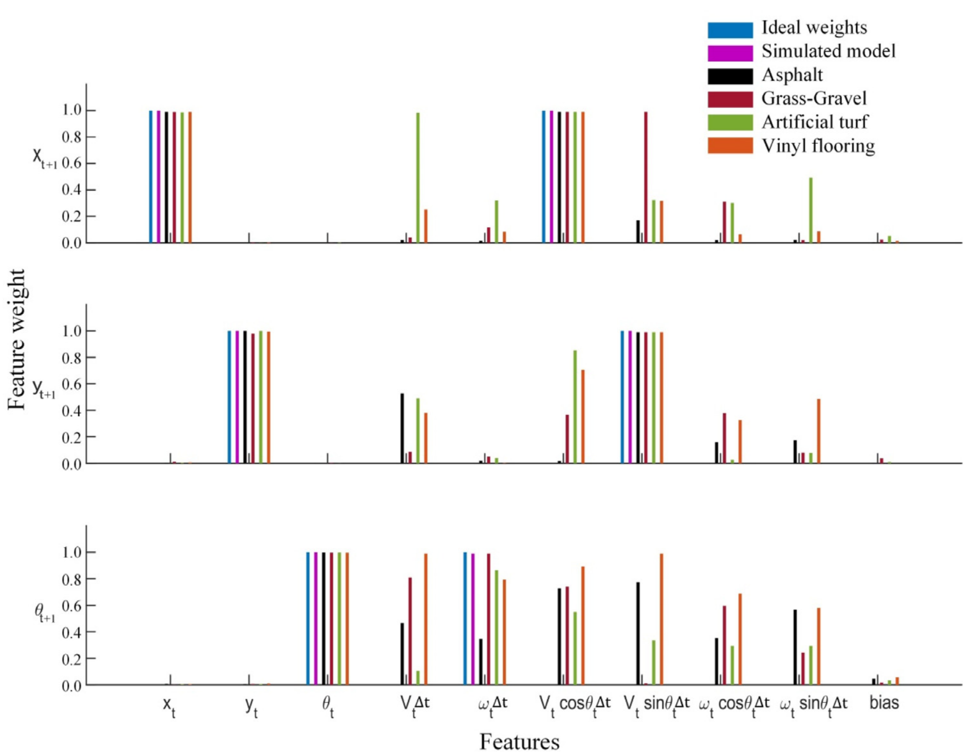

Fig. 2. The absolute weights corresponding to each feature as obtained from the least squares fit.

It is possible that some of the above features may not contain characteristic information about the terrain, and as such may not contribute much for the terrain estimation application. It is also possible that by choosing all features with weights > 0.5 in at least one terrain condition, some of the weights could be redundant. However, these problems will be addressed through the use of PCA over the estimated weights, as explained later in Section 3.3.

The model given in Eq. (4) can be used in the real-time estimation of the weights, provided the pose data (x, y, θ) of the robot at time t and $t + \Delta t$, along with the control input (V, ω) is available. The estimated value of the weights in turn can be used to identify the terrain over which the robot moves. These points will be explored in greater detail in the following section.

3.2. Real-time estimation of weights

Based on the generalized model given in Eq. (4), the following section will explain the use of a moving window least squares estimator to obtain the best fitting value of the weights corresponding to each feature in real-time. As mentioned in the previous section, at the least, a noisy estimate of the robot's pose is necessary to perform a least square estimation of the weights. This is inherently necessary information for any autonomous navigation application and can be obtained from different pose estimation sensors such as the POZYX [27], LOSA [31], real-time kinematic (RTK) GPS, visual odometry etc. For the purpose of this paper we will be analyzing data obtained from the POZYX system.

Based on Eq. (4), Eq. (2) can be modified as;

$$\mathbf{Y} = \mathbf{w}\mathbf{x} \tag{5}$$

It should be noted that while w is still a 3×9 matrix, only 15 weights are non-zero (four for x_{t+1}, four for y_{t+1}, and seven for θ_{t+1}) as denoted by Eq. (4). In theory, all of the unknown weights could be estimated accurately with just 15 measurements, but there are some practical limitations to this. The POZYX system or any other positioning system working in outdoor environments is susceptible to noise. As such, using the minimum number of measurements will result in a noisy estimation of the unknown weights. Therefore, a large number of measurements needs to be used with a moving window least squares estimator to reduce the effect of noise and improve accuracy. On the other hand using a very large window will require a large set of prior measurements before the weights can be estimated. It will also lead to high computational cost. Moreover, with a large enough window it is possible to filter out the unique effects of the terrain, which could adversely affect the terrain estimation applications that will be discussed later. For the purpose of this paper the filter size was manually tuned to 400 samples to yield the best performance. It should be noted that the moving window starts the estimation only after receiving the first 400 samples which takes about four seconds based on the current setup.

3.3. Application to terrain estimation

This section describes the use of support vector machine (SVM) [32] in performing terrain classification based on the estimated value of the weights. As given by Eq. (4), the estimator returns 15 weights at each instance, but it is possible that the some of them are interrelated, resulting in redundancy of information.

Principal component analysis is a popular method used in machine learning community to remove redundancy of information. The principal components are a linear combination of the original variables that are orthogonal to each other. For the proposed application, even though the total number of PCs is the same as the number of estimated weights, the first ten PCs are sufficient to capture more than 98% of the data. It should be noted that, PCA is essentially eigenvalue decomposition and therefore the chosen PCs are the first ten eigenvectors of the covariance matrix of the estimated weights, in decreasing order of eigenvalue.

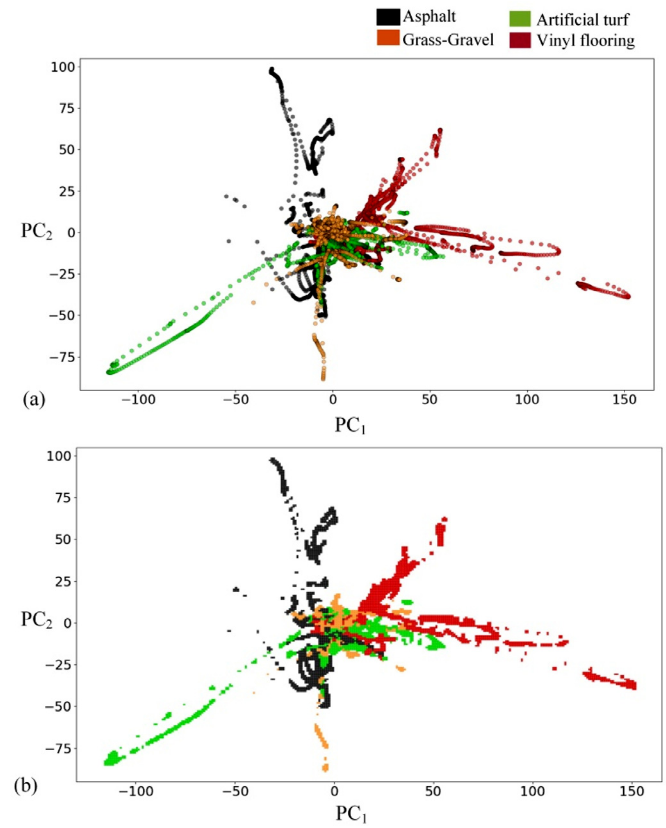

Fig. 3. The first two PCs of the estimated weights. The figure shows (a) ground truth and (b) the classification results from the SVM. The magnitude of the first PC is along the X axis and the second along the Y axis.

Using the orthogonal transformation given by the first ten PCs, the estimated values of the 15 weights w_{ij} as given by Eq. (4) were transformed into reduced dimensional data along the PCs w_{pi} $(i = 1 - 10)$. This reduced dimensional data transformed along the PCs, hereafter referred to as the PCs of the estimated weights, was used in performing terrain estimation. By using the PCs of the estimated weights for the terrain estimation, the computational load in performing the estimation is reduced while maintaining an acceptable amount of accuracy.

SVMs have been used previously in terrain estimation, specifically on vibration based terrain estimation techniques [17,18]. They allow for nonlinear classification based on the kernel function used inside the SVM. For the proposed terrain estimation technique an SVM with a Radial Basis kernel function (RBF) was used on the reduced dimensional data. For the SVM implementation we used the Statistics and Machine Learning Toolbox in MATLAB [33]. A *one- versus-rest classification* was implemented using multiclass error-correcting output codes (ECOC) model containing multiple SVM binary learners as provided by the MATLAB Toolbox. Before training the SVM, the collected data was randomly divided into training and test data sets in 3:1 ratio.

Fig. 3 shows the first two PCs of the collected data with ground truth marking and the SVM terrain estimation over the whole range of values. Using ten PCs gives 85.27% accuracy on the training data and 84.3% accuracy on the test data. The confusion matrix showing the performance of the trained SVM is shown in Fig. 4. Based on the results provided, the proposed method performs well for terrain estimation application. It is important to note that, even though the exact value of the weights themselves could depend on the terrain as well as on the characteristics of the robot including the type of tracks, treads used on the track, presence or absence of suspension system and their layout. By using labeled data collected with the same robotic platform over different terrain conditions, the trained SVM is able to disregard the common factors namely

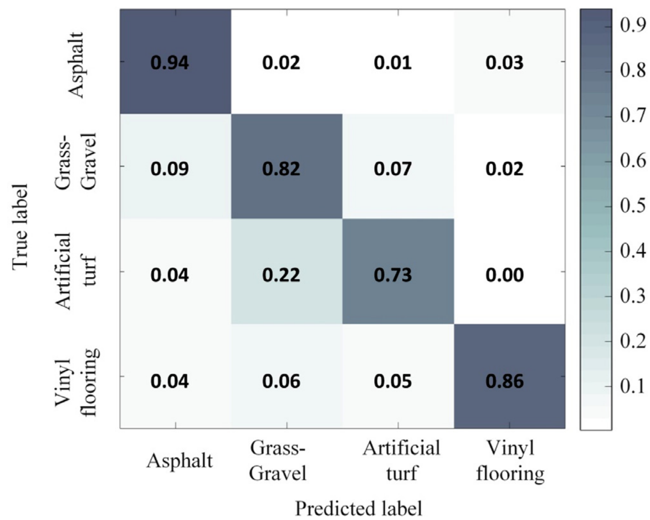

Fig. 4. Normalized confusion matrix obtained from the trained SVM terrain classification.

the characteristics of the robot, and instead focus on the terrain relevant features for performing better estimation.

Fig. 5 summarizes the overall approach, including all the steps for the online terrain estimation and the offline training that was described in this section, along with the results that are obtained at each intermediate step. The offline steps for collecting the training data, estimating the features of interest, estimating the principal components from the training data, and the training of the SVM itself needs to be performed only once. These steps have to be repeated only if the locomotion model of the robot changes, such as in the case of using a car-like robot or if additional terrain categories need to be added. In contrast to solid mechanics or finite-element modeling methods the data driven approach presented here allows the SVM to learn distinct terrain types based on the labeled data provided during the offline training phase. Even though the a-priori training is computationally intensive, it is performed offline, whereas the computationally efficient estimation can be run in real-time.

4. Validation of the proposed approach

The performance of the trained SVM for real-time terrain estimation applications was validated using simulated terrain transition data, generated from the collected real-world training data. Even though the training data was collected by driving the robot on four uniform terrain conditions separately, the collected data samples from different terrain conditions could be stacked together to simulate robot motion over varying terrain conditions. It should be noted that an individual data sample consists of pose data (x, y, θ) of the robot at time t and $t + \Delta t$, along with the control input (V, ω). All such data samples collected when the robot moves over a single terrain condition is referred to as dataset for a single terrain type in the following discussion.

The moving window least squires estimation of the weights requires 400 consecutive samples from a single terrain type to perform accurate estimation of the weights. Stacking random individual samples from different terrain conditions could result in unrealistic estimates of the weights. The SVM on the other hand looks at each individual sample of the ten PCs to produce terrain estimations. In order to validate the performance of the proposed technique in detecting terrain transitions, we artificially generated test cases for terrain transitions using the real-world data. The weights were estimated from each dataset corresponding to a single terrain type and the PCs were obtained. 500–600 samples of the PCs were then selected from each individual terrain type and stacked together in a random order such that it would appear as if the robot was transitioning from one terrain condition to another for the trained SVM. The trained SVM was used to estimate the terrain condition from the stacked PCs of the estimated weights and compared with the ground truth marking on the dataset. The results are shown in Fig. 6.

It is clear from the results that the proposed approach has the ability to recognize terrain conditions with a high degree of accuracy, 93.84%, with fast detection of terrain transitions. It should be noted that instead

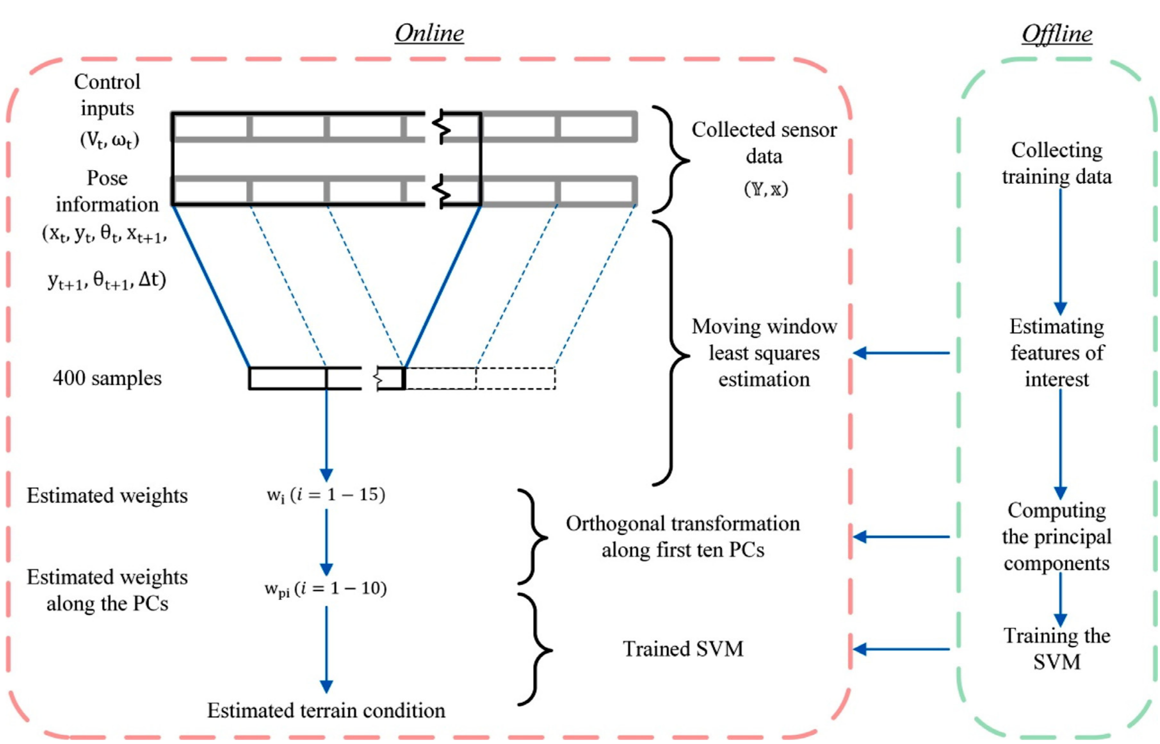

Fig. 5. Flow chart showing the working of the proposed terrain estimation algorithm.

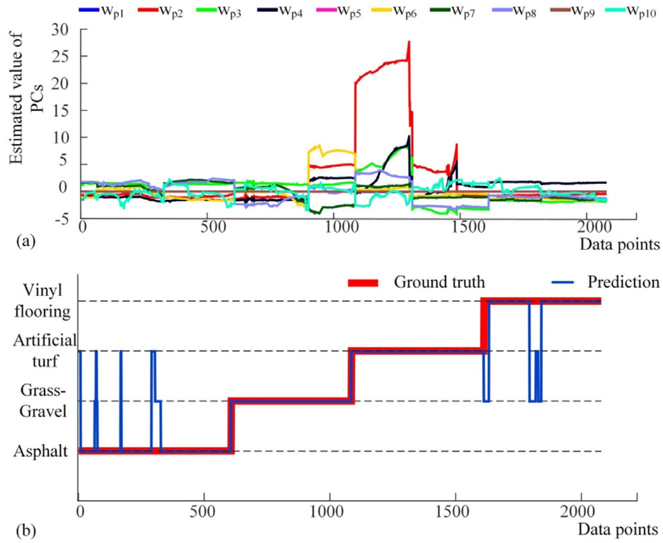

Fig. 6. Validation of the trained SVM over simulated terrain transition data with (a) estimated values of the PCs used by the SVM and (b) the terrain estimation results from the SVM.

of state evolution data, the PCs of the estimated weights were given directly to the SVM. As such, the simulation experiments do not reflect the delay caused by the filter.

In order to validate the proposed terrain estimation technique, two different experimental cases were considered. In each case, the robot was made to run on a region consisting of two different types of terrain. The first case being asphalt and vinyl concrete flooring and the second vinyl concrete flooring and grass–gravel. It should be noted that although the experimental validation was performed on the same types of terrain as in the data collection, it was not performed in the same location. The experimental validation required regions where two different terrain conditions were present within the six meter square sensing limits of the POZYX positioning system. On the other hand, training data was collected in regions where the terrain conditions were uniform throughout.

The two experimental setups are shown in Fig. 7. For the experimental validation described here, the robot was manually driven between the different terrains at relatively low velocity of 0.2–0.5 m/s. As in the data collection phase, the POZYX system was set up to record the pose of the robot at 100 Hz sampling rate. The manual control inputs along with the state evolution information were used to estimate the weights. Based on the PCs of the estimated value of the weights the terrain classification was performed in real-time. The experimental platform, STORM, has an ODROID XU4 computer for onboard processing. Due to the limited computational capability of the onboard computer, the collected data was sent via ROS [34] to an HP laptop with a 2.6 GHz Intel processor and 8GB RAM. The real-time estimation of the weights and the terrain classification based on the PCs of the estimated weights were performed on the laptop. Figs. 8 and 9 show the PCs of the estimated value of the weights used by the SVM along with the estimation results. It should be noted that for both experiments the transition boundary is a straight line, as shown in Fig. 7. The POZYX absolute positioning system was setup such that the world coordinate system had one axis parallel to the terrain transition boundary. This allows for easier estimation of ground truth terrain condition of the robot.

The position coordinates of the boundary was measured and recorded prior to the experiment. For cases when the robot position was less than the recorded value, the ground truth was marked as terrain 1. Similarly, when the robot position was greater than the recorded value, the ground truth was marked as terrain 2. It should be noted that the above setup was used solely for ease of marking the ground truth

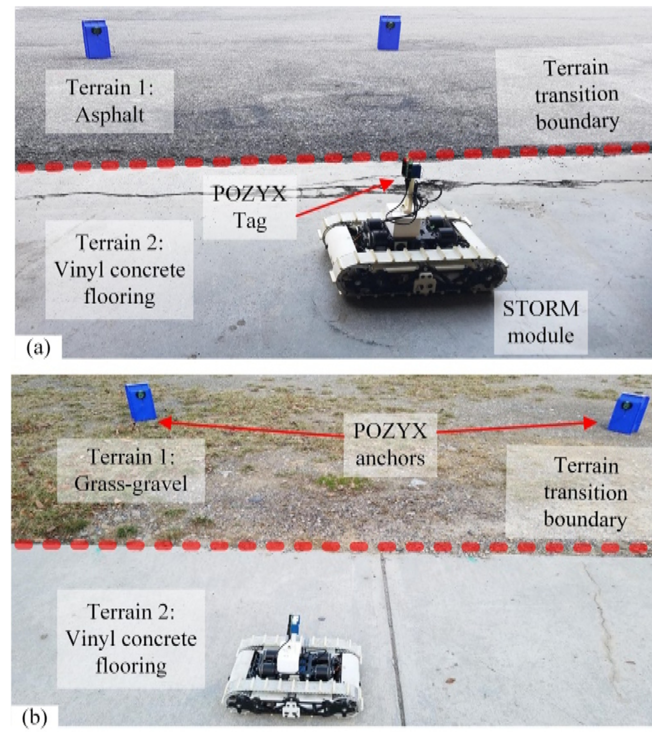

Fig. 7. Experimental setup for the validation of the proposed terrain estimation technique on two different cases: (a) asphalt and vinyl concrete floor, (b) vinyl concrete flooring and grass–gravel.

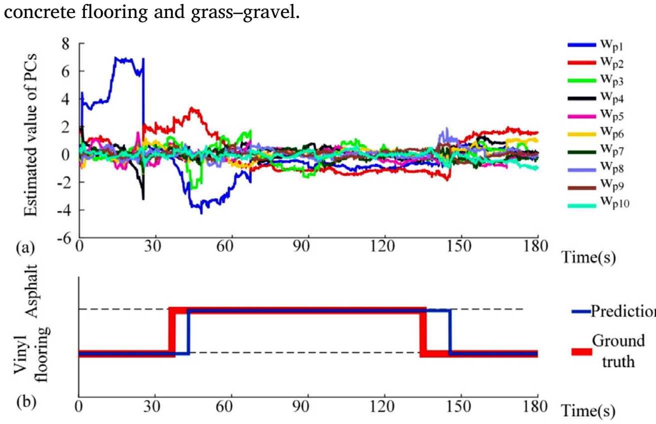

Fig. 8. Validation of the trained SVM through experiments on asphalt and vinyl flooring with (a) PCs of the estimated values of the weights used by the SVM and (b) terrain estimation results along with ground truth.

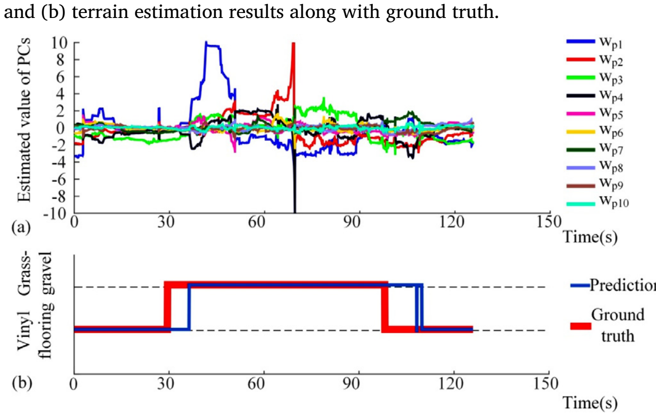

Fig. 9. Validation of the trained SVM through experiments on vinyl flooring and grass–gravel with (a) PCs of the estimated values of the weights used by the SVM and (b) terrain estimation results along with ground truth.

Table 2
Comparison of proposed technique with existing state-of-the-art techniques.

Sensing modality	Output	Accuracy	Disadvantage
Radar [7]	Binary detection of obstacles	100% when used in combination with LIDAR	Cannot be used as a standalone sensor
Monocular camera [8]	Binary detection of traversability	98.53% at pixel level prediction	Subject to failure of vision sensors such as in presence of fog, smoke or dust
Stereo camera [9]	Binary detection of traversability	92% classification accuracy	
Measured soil and wheel parameters with classical terra-mechanics equations [11]	Terrain parameter estimation for binary detection of traversability	–	Need additional sensors for estimating normal force and torque on the wheel, wheel sinkage etc
Vibration induced through wheel-terrain interaction [13]	Online identification of terrain: sand, gravel, and clay	96% on sand, 78% on dirt, and 82% on gravel	May not be applicable to tracked vehicles as the vibration experienced by the tracks are influenced by weight of vehicle and properties of suspension system
Intelligent tire [21]	Classification between asphalt, concrete, soil, and grass	–	
Accelerometer attached on a tactile probe [14]	Classification between ten different indoor and outdoor surfaces	94.6% success	Limits mobility of the vehicle
Actuator and inertial sensor data from vehicle [19]	Classification between snow linoleum, ice, deep water	90%	Designed specifically for rotating leg mechanisms
Proposed technique of using state evolution for known control commands	Classification between asphalt, grass–gravel, vinyl and artificial turf	90.53% on asphalt and vinyl pair, 85.33% accuracy on vinyl and grass–gravel pair.	No dedicated sensors needed. Capable of working on any traversable terrain

and is not a limitation of the proposed terrain estimation method, as it can detect terrain transition boundaries of any shape. The ground truth, marked based on the position of the robot, is also overlaid with the estimation results. Both the ground truth and the estimations have been shown only after the first 400 samples have been taken by the estimator after the robot started moving.

For the experimental data, the SVM accuracy came out to be 90.53% on asphalt and vinyl pair, 85.33% accuracy on vinyl and grass–gravel pair. The lower accuracy of the SVM over the experimental data could be attributed to the fact that it is operating on the PCs of the weights that are being estimated in real-time using the moving window approach, as compared to the simulation case where the computed PCs were directly passed into the SVM. As mentioned before, at any given instant, the moving window estimator looks at 400 samples from the past for estimating the weights. This means that it takes at least four seconds (time taken to collect 400 data samples) after the robot has moved into the new terrain condition for the estimated weights to reflect the change accurately. This is evident in Fig. 8, when the SVM does not immediately recognize when the robot goes back from the asphalt into vinyl flooring condition. Similar inference can be drawn from Fig. 9. In both cases it takes about 8–15 s for the SVM to accurately predict the terrain condition after the robot has made the transition. Even after the robot has moved onto the new terrain, it has to move around and collect sufficient data before the estimated weights reflect the change to the extent that the SVM architecture can make accurate predictions. The rest of the time delay could be associated with the time taken for the value of the estimated weights to change, which is dependent on the terrain condition, the robot, and the motion commands being sent to the robot.

Prior knowledge about the region having only two kinds of terrain was leveraged by constraining the classification results to only the candidate regions for the duration of the experiment. For example in the vinyl and grass–gravel case, the SVM only chose between vinyl and grass–gravel based on whichever had the best score. If all four terrain types are included in the classification, it could lower the accuracy of the classification results as suggested by the confusion matrix shown in Fig. 4 and the simulated terrain transition results shown in Fig. 6. The results presented in Figs. 8 and 9 provide proof-of-concept validation of the idea that terrain estimation can be performed solely based on the state evolution of the robot. Detailed discussion on the potential ways to improve the terrain classification accuracy is provided in Section 6.

It should be noted that since only the PCs of the estimated weights are passed into the SVM and not the x and y coordinates of the robot, the classifier does not learn the coordinates of the regions that correspond to each terrain condition. This ensures that the learned architecture is generalized such that it could be applied when the robot moves over previously unseen terrain conditions.

A summary of state-of-the-art techniques using different sensing modalities for performing terrain classification and traversability estimation is given in Table 2. The output from each proposed technique, their reported accuracy, and possible disadvantages are also given along with the proposed method for comparison. A quantitative comparison of the proposed technique with existing state-of-the-art terrain estimation techniques is not provided as it depends on many additional factors including the specifications of the sensor used for collecting data, computational capabilities of the experimental setup, etc.

As the proposed technique does not require any dedicated sensor, a direct quantitative comparison of accuracy is not possible. From Table 2 it can be inferred that the proposed technique performs at least as good as the state-of-the-art techniques in terms of the number of detected terrain conditions and accuracy. As mentioned previously, the major advantage of the proposed technique is that it does not require any dedicated sensing modality which in turn removes associated failure modes, costs, and computational overload. Multi sensor techniques could be used for overcoming the limitations of each individual sensing modality, but this comes with the added cost of sensors and computational overhead. On the other hand, the proposed technique requires only the state evolution of the robot along with the control commands performed by the robot, both readily available on all autonomous or semi-autonomous mobile robots. It should be noted that while this work reports proof of concept validation of the proposed method, its performance in terms of accuracy and number of detected terrain conditions can be improved with more data.

5. Proposed application to autonomous navigation

Real-time terrain estimation can be used to improve autonomous navigation, specifically in control, autonomous path planning, and in improving robot localization. It should be noted that while the techniques proposed in this section have the potential to improve autonomous navigation, experimental validation of these techniques is beyond the scope this work and will be addressed as part of future work.

Varying nature of terrain, specifically slip can result in varying trajectory tracking performance of the robot, unless the trajectory tracking controller adjusts the gains according to the nature of the terrain. Previously an extended Kalman filter based approach toward handling this issue was proposed in [35]. Assuming optimal set of gains for each terrain condition is available in the form of a look up table, real-time terrain estimation can enable the robot to choose the optimal gains for each terrain condition and thereby guarantee optimal performance. Since the terrain estimation approach only relies on the control output (V_t, ω_t) and the state evolution of the robot, the changing gain values on the trajectory tracking controller will not affect the terrain estimation process.

The proposed real-time terrain estimation techniques can be used to improve path planning for field robotic applications such as in search and rescue scenarios. As of today, the reported field trails [1,2] have primarily relied on the intuition of the remote operator, based on visual feedback, to decide whether a robot should favor a particular terrain over another. This could be automated by monitoring the estimated values of the weights associated with the features of interest. Based on the deviation of the estimated weights from the ideal unicycle model, the relative traversability of multiple terrain conditions could be compared. Terrains that deviate more would be less traversable as compared to terrains that produce closer to ideal weights. For instance, note that in Fig. 2 the weights corresponding to features having control inputs (V_t and ω_t terms) are high for x_{t+1}, and y_{t+1}. But for θ_{t+1}, the weights are relatively low. Based on this, it can be inferred that it is difficult to turn the robot on asphalt. This was noticed during the experiment as the minute cracks on the asphalt provide greater resistance to turning as the wedges on the track become entrenched in them. Similar inferences can be drawn for other terrain conditions as well. This information could then be used in autonomous path planning applications by assigning a relatively high cost for less traversable terrain conditions. This allows for a more complete sense of autonomy for rescue robotic systems, with lesser reliance on human input. Even in the absence of absolute positioning sensors, such as GPS or POZYX, which is usually the case with disaster scenarios, the weights can be estimated by relying on visual odometry. Moreover, since this does not require classifying the terrain into previously learned terrain types using the SVM, the proposed techniques could be applied on previously unseen terrain conditions. It should be noted that for the proposed technique to estimates the weights, the robot must be able to traverse the terrain for a period of time. If the terrain condition causes the robot to immediately get stuck, the proposed technique would fail.

In addition to the above mentioned applications, terrain estimation can also be used in localization and mapping, during remote sensing or while surveying disaster scenarios. This could be of particular importance when the robot is moving over unstructured terrain conditions such as underground mines, as in the new DARPA challenge [6]. These regions have no GPS reception, and relying purely on visual odometry methods could lead toward accumulating drift error. On the other hand with real-time terrain recognition, the terrain itself could be used as landmarks, such that coupled with visual odometry methods precise localization of the robot can be done.

6. Conclusion

This work aimed at using state evolution of a robot moving over varying terrain conditions, under known control inputs, to perform real-time terrain estimation. The weights in the governing equation for robot motion were estimated from the state evolution data which was then transformed along the PCs and passed on to a trained SVM to generate the real-time terrain estimates. The proposed approach was validated over experimental data collected from four different terrain conditions. The trained SVM reported accuracy over 80% for simulated and actual testing.

This paper presents a proof-of-concept validation of the idea that terrain estimation can be performed solely based on the state evolution of the robot. Although the experimental validity of the proposed work was demonstrated using the POZYX system, it should be noted that the proposed method does not depend on any specific sensing technique. In fact, the experimental validation shows that the method works well with the limited range and noisy output of POZYX. Using a better pose estimation system such as the LOSA [31], RTK GPS or Differential GPS (DGPS) is expected to significantly improve the performance of the proposed technique. Detailed analysis on the effect of the various features of interest and their associated weights on the proposed terrain classification technique will be analyzed as part of future work along with detailed experimentation using various position sensing modalities over a wide variety of terrain and ambient conditions. Further testing with different kernels for the SVM or using other machine learning techniques to improve the estimation performance will also be explored in future. In addition, as mentioned in Section 4, the robot velocity was kept low for the data collection as well as the experiments. Recording robot motion with a wider range of velocities on different terrain conditions can result in a richer dataset, allowing the trained SVM to perform better with various terrain conditions.

In addition to the estimated value of the weights, future work will look into using additional features of the data including the variance of the collected parameters, skewness, kurtosis etc., for information about the terrain. Adding these additional features is expected to improve the accuracy of the estimation method based on a preliminary analysis of the collected data. It should be noted that the as per the above analysis, the estimated value of the weights can depend on the features of the mobility platform, as much as they depend on the terrain over which the robot is moving. This in turn opens up possibilities for using the above method as a diagnostic tool in order to gauge the performance of the robot. Assuming that the terrain conditions remain the same, changes in the estimated value of the weights can be used to determine the presence of faults in the robot. For all of the tests conducted as part of this paper the performance of the mobility platform was assumed to be same, but future analysis will focus on using the proposed method for fault detection in the robot.

Even though this paper explored the use of terrain estimation methods as applied to a skid steer robot, ongoing work aims at applications on commercial vehicles such as cars and trucks. The proposed technique can be used to estimate driving conditions such as presence of snow or ice on the road and also toward monitoring the conditions of the vehicle in real-time. The presence of suspension systems in these vehicles can adversely affect the performance of terrain estimation methods as it reduces the effect of the terrain on the motion of the vehicle. This may require attaching sensors directly to the wheels of the vehicle, before the suspension system, as explored in [36].

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.

References

[1] Murphy RR, Tadokoro S, Kleiner A. Disaster robotics. Springer handb. robot.. Springer International Publishing; 2016. p. 1577–604. doi:10.1007/978-3-319-32552-1_60.

[2] Murphy R, Kravitz J, Stover S, Shoureshi R. Mobile robots in mine rescue and recovery. IEEE Robot Autom Mag 2009;16(2):91–103. doi:10.1109/MRA.2009.932521.

[3] Casper J, Murphy RR. Human-robot interactions during the robot-assisted urban search and rescue response at the World Trade Center. IEEE Trans Syst Man, Cybern Part B 2003;33(3):367–85. doi:10.1109/TSMCB.2003.811794.

[4] Thompson K. Squad multipurpose equipment transport (SMET) 2015. https://ndiastorage.blob.core.usgovcloudapi.net/ndia/2016/GRCCE/Thompson.pdf (accessed December 1, 2018).
[5] Massey K. Squad mission equipment transport (SMET): lLessons learned for industry 2016. https://ndiastorage.blob.core.usgovcloudapi.net/ndia/2016/GRCCE/Massey.pdf (accessed December 1, 2018).
[6] DARPA Subterranean (SubT) Challenge 2017. https://www.darpa.mil/program/darpa-subterranean-challenge (accessed November 6, 2018).
[7] Johnston. Off-highway obstacle detection. IEEE Instrum Meas Mag 2006;9(6):16–24. doi:10.1109/MIM.2006.1708345.
[8] Mei J, Yu Y, Zhao H, Zha H. Scene-Adaptive off-road detection using a monocular camera. IEEE Trans Intell Transp Syst 2018;19(1):242–53. doi:10.1109/TITS.2017.2768573.
[9] Bellone M, Reina G, Caltagirone L, Wahde M. Learning traversability from point clouds in challenging scenarios. IEEE Trans Intell Transp Syst 2018;19(1):296–305. doi:10.1109/TITS.2017.2769218.
[10] Sebastian B, Ben-Tzvi P. Physics based path planning for autonomous tracked vehicle in challenging terrain. J Intell Robot Syst 2018 Available online. doi:10.1007/s10846-018-0851-3.
[11] Iagnemma K, Kang S, Shibly H, Dubowsky S. Online terrain parameter estimation for wheeled mobile robots with application to planetary rovers. IEEE Trans Robot 2004;20(5):921–7. doi:10.1109/TRO.2004.829462.
[12] Reina G, Ojeda L, Milella A, Borenstein J. Wheel slippage and sinkage detection for planetary rovers. IEEE/ASME Trans Mechatron 2006;11(2):185–95. doi:10.1109/TMECH.2006.871095.
[13] Brooks CA, Iagnemma K. Vibration-based terrain classification for planetary exploration rovers. IEEE Trans Robot 2005;21(6):1185–91. doi:10.1109/TRO.2005.855994.
[14] Giguere P, Dudek G. A simple tactile probe for surface identification by mobile robots. IEEE Trans Robot 2011;27(3):534–44. doi:10.1109/TRO.2011.2119910.
[15] Park B, Kim J, Lee J. Terrain feature extraction and classification for mobile robots utilizing contact sensors on rough terrain. Procedia Eng 2012;41:846–53. doi:10.1016/j.proeng.2012.07.253.
[16] Wolf DF, Sukhatme GS. Semantic mapping using mobile robots. IEEE Trans Robot 2008;24(2):245–58. doi:10.1109/TRO.2008.917001.
[17] Weiss C, Frohlich H, Zell A. Vibration-based terrain classification using support vector machines. In: 2006 IEEE/RSJ international conference. on intelligent robots and systems. IEEE; 2006. p. 4429–34. doi:10.1109/IROS.2006.282076.
[18] Weiss C, Fechner N, Stark M, Zell A. Comparison of different approaches to vibration-based terrain classification. In: 3rd European conference on mobile robots; 2007. p. 7–12.
[19] Giguere P, Dudek G, Saunderson S, Prahacs C. Environment identification for a running robot using inertial and actuator cues. Robot science. and systems II Philadelphia, PA. Robotics: Science and Systems Foundation; 2006. doi:10.15607/RSS.2006.II.035.
[20] Wang S, Kodagoda S, Shi L, Dai X. Two-Stage road terrain identification approach for land vehicles using feature-based and Markov random field algorithm. IEEE Intell Syst 2018;33(1):29–39. doi:10.1109/MIS.2017.2581327.
[21] Khaleghian S, Taheri S. Terrain classification using intelligent tire. J Terramechanics 2017;71:15–24. doi:10.1016/j.jterra.2017.01.005.
[22] Khaleghian S, Emami A, Taheri S. A technical survey on tire-road friction estimation. Friction 2017;5(2):123–46. doi:10.1007/s40544-017-0151-0.
[23] Kumar A, Ben-Tzvi P. Novel wireless sensing platform for experimental mapping and validation of ship air wake. Mechatronics 2018;52:58–69. doi:10.1016/j.mechatronics.2018.04.009.
[24] Karpatne A, Atluri G, Faghmous JH, Steinbach M, Banerjee A, Ganguly A, et al. Theory-Guided data science: a new paradigm for scientific discovery from data. IEEE Trans Knowl Data Eng 2017;29(10):2318–31. doi:10.1109/TKDE.2017.2720168.
[25] Rudy SH, Brunton SL, Proctor JL, Kutz JN. Data-driven discovery of partial differential equations. Sci Adv 2017;3(4):e1602614. doi:10.1126/sciadv.1602614.
[26] Kumar P, Saab W, Ben-Tzvi P. Design of a multi-directional hybrid-locomotion modular robot with feedforward stability control. ASME international design engineering technical conference on computer and informatics engineering conference (IDETC/CIE) Cleveland, Ohio. ASME; 2017. Aug. 6-9. doi:10.1115/DETC2017-67436.
[27] Pozyx Labs. Pozyx accurate positioning 2015. https://www.pozyx.io/ (accessed June 25, 2018).
[28] Martínez JL, Mandow A, Morales J, Pedraza S, García-Cerezo A. Approximating kinematics for tracked mobile robots. Int J Rob Res 2005;24(10):867–78. doi:10.1177/0278364905058239.
[29] Fink JR, Stump EA. Experimental analysis of models for trajectory generation on tracked vehicles. In: IEEE international conference on intelligent robots and systems. IEEE; 2014. p. 1970–7. doi:10.1109/IROS.2014.6942824.
[30] Mandow A, Martínez JL, Morales J, Blanco JL, García-Cerezo A, González J. Experimental kinematics for wheeled skid-steer mobile robots. In: IEEE international conference on intelligent robots and systems. IEEE; 2007. p. 1222–7. doi:10.1109/IROS.2007.4399139.
[31] Kumar A, Ben-Tzvi P. Spatial object tracking system based on linear optical sensor arrays. IEEE Sens J 2016;16(22):7933–40. doi:10.1109/JSEN.2016.2607120.
[32] Cortes C, Vapnik V. Supprot-vector networks. Mach Learn 1995;297(20):273–97.
[33] MathWorks. Statistics and machine learning toolbox. Matlab; 2006 https://www.mathworks.com/help/stats/index.html (accessed January 10, 2019).
[34] Open Source Robotics Foundation. Indigo - ROS Wiki 2014. http://wiki.ros.org/indigo (accessed July 7, 2018).
[35] Sebastian B, Ben-Tzvi P. Active disturbance rejection control for handling slip in tracked vehicle locomotion. J Mech Robot 2019;11(2):021003. doi:10.1115/1.4042347.
[36] Singh KB, Ali Arat M, Taheri S. An intelligent tire based tire-road friction estimation technique and adaptive wheel slip controller for antilock brake system. J Dyn Syst Meas Control 2013;135(3):031002. doi:10.1115/1.4007704.

Bijo Sebastian received his B.Tech. degree in Mechanical Eng. from College of Engineering, Trivandrum, India in 2013. He received his M.Tech. in Mechatronics from Central Mechanical Engineering Research Institute, West Bengal, India in 2015. He is currently pursuing Ph.D. at the Virginia Tech under the guidance of Dr. Pinhas Ben-Tzvi in developing autonomous navigation capabilities for mobile robots in unstructured terrain conditions with application toward casualty evacuation (CASEVAC). In addition to localization, planning, and control of mobile robots, he has also worked on design and control of exoskeleton gloves.

Pinhas Ben-Tzvi (S'02–M'08–SM'12) received the B.S. degree (summa cum laude) in mechanical engineering from the Technion—Israel Institute of Technology, Haifa, Israel, and the M.S. and Ph.D. degrees in mechanical engineering from the University of Toronto, Toronto, ON, Canada. He is currently an Associate Professor of Mechanical Engineering and Electrical and Computer Engineering, and the founding Director of the Robotics and Mechatronics Laboratory at Virginia Tech, Blacksburg, VA, USA. His current research interests include robotics and intelligent autonomous systems, mechatronics, human-robot interactions, dynamic systems and control, mechanism design and system integration, and novel sensing and actuation.
Ben-Tzvi is the recipient of the 2018 Virginia Tech Faculty Fellow Award, the 2013 GW SEAS Outstanding Young Researcher Award and the GW SEAS Outstanding Young Teacher Award, as well as several other honors and awards. Dr. Ben-Tzvi is Technical Editor for the IEEE/ASME Transactions on Mechatronics, Associate Editor for ASME Journal of Mechanisms and Robotics, Associate Editor for IEEE Robotics and Automation Magazine, and an Associate Editor for the Int'l Journal of Control, Automation and Systems and served as an Associate Editor for IEEE ICRA 2013–2018. He is a member of the American Society of Mechanical Engineers (ASME).