







On the role of feature and signal selection for terrain learning in planetary exploration robots

Angelo Ugenti¹  | Fabio Vulpi^{1,2}  | Raúl Domínguez³  | Florian Cordes³  |
Annalisa Milella²  | Giulio Reina¹ 

¹Department of Mechanics, Mathematics and Management, Polytechnic of Bari, Bari, Italy

²Institute of Intelligent Industrial Technologies and Systems for Advanced Manufacturing, National Research Council, Bari, Italy

³DFKI, Robotics Innovation Center, Bremen, Germany

Correspondence

Angelo Ugenti, Department of Mechanics, Mathematics and Management, Polytechnic of Bari, Via Orabona 4, 70125 Bari, Italy.
Email: angelo.ugenti@poliba.it

Funding information

Horizon 2020 Framework Programme

Abstract

Increasing the terrain awareness of planetary exploration rovers is one key technology for future space robotics to successfully accomplish long-distance and long-duration missions. In contrast to most of the existing algorithms that use visual or depth data for terrain classification, the approach presented in this study tackles the problem using proprioceptive sensing, for example, vibration or force measurements. The underlying assumption is that these signals, being directly modulated by the terrain properties, are well descriptive of a given surface. Therefore, terrain signature can be inferred via learning algorithms that are trained on either the signals directly or a signal-derived feature set. Following the latter approach, first, a physics-based signal augmentation process is presented that aims at maximizing the information content. Then, a feature selection algorithm based on a scoring system and an iterative search is developed to decrease the computational cost while preserving high classification accuracy. The resulting most informative feature subspace can be used to train a support vector machine (SVM) classifier. For comparison, the time histories of the selected proprioceptive signals are used to train a deep convolutional neural network (CNN). Results obtained from real experiments using the SherpaTT rover confirm that proprioceptive sensing is effective in predicting terrain type with an accuracy higher than 90% for both algorithms in generalization tasks. When the two learning approaches are contrasted in extrapolation problems, for example, predicting observations acquired at previously unseen velocity or terrain, CNN outperforms the standard SVM. Furthermore, CNN holds the additional advantage of learning features automatically from signal spectrograms, reducing the need of a priori knowledge at the expense of higher computational efforts.

KEYWORDS

deep learning, feature selection, learning methods, planetary exploration robots, proprioceptive sensing, terrain classification, vehicle-terrain mechanics

1 | INTRODUCTION

This study has been developed as part of the research activity for the project ADE (Autonomous DEcision making in very long traverses) (Ocón et al., 2020), funded by the European Union's Horizon 2020 research and innovation programme. The main goal of ADE is to

develop and test a rover system capable to achieve autonomous long-range navigation in hostile environments while guaranteeing consistent data collection. The mobility range of planetary exploration rovers has been up to date limited to a few hundred meters per sol day (ESA, 2021; JPL, 2021; NASA, 2020). From a purely technical point of view, this limitation has both hardware and software sources.

The former and most important is the finite power storage of the rover locomotion system, which is fixed given a robot design. The latter is reduced skills in terms of autonomous decision-making, that can be improved by artificial intelligence. Improving these capabilities extends the autonomy of the rover across multiple geographical areas and therefore expands opportunities of data collection.

Directly related to long-range navigation is also a safety issue. The importance of sensing hazards was highlighted, for example, in April 2005, when the Mars exploration rover Opportunity became embedded in a dune of loosely packed drift material (Cowen, 2005). The terrain geometry as reconstructed from a distance via stereovision did not indicate any hazard. However, the high compressibility of the loose drift material caused the wheels to sink deeply into the surface. The combination of the drift's low internal friction and the motion resistance due to sinkage prevented the rover from producing sufficient thrust to travel up the slope. Opportunity's progress was delayed for more than a month while engineers worked to find a way out. A similar embedding event led to the end of operations for the twin rover Spirit in 2010.

Therefore, future generations of planetary exploration rovers will require key technologies suitable to overcome these limitations, performing long traverses while guaranteeing fast reaction, mission reliability and safety, and optimal exploitation of the robot's resources within reasonable costs.

In this context, the ability to sense and characterize the incoming terrain would represent an enabling technology towards long-term autonomy and potential hazard avoidance (Nampoothiri et al., 2021).

The objective of this paper is to demonstrate the potential of terrain classification via learning algorithms that are trained on proprioceptive features. Here, proprioceptive features refer to statistics that are extracted from the measurement of a physical variable pertaining to the robot-environment interaction, for example, wheel velocity, forces, body linear, and angular accelerations.

The hypothesis is that being modulated by the terrain properties, these features are a rich source of information from which the specific terrain type can be inferred via learning approaches (Brooks & Iagnemma, 2005; Gonzalez et al., 2019).

One of the contributions of this study refers to the selection of the most informative subset of proprioceptive features derived from the sensor suite integrated onboard of planetary exploration rovers. A range of aspects is addressed that includes feature extraction, feature ranking, multivariate feature selection, and efficient feature space construction. While feature selection has been largely investigated in other domains, for example, image processing, text processing, and gene expression analysis (Guyon & Elisseeff, 2003), it remains largely under-investigated for the terrain classification problem of planetary exploration rovers, and rough-terrain robots, in general. In contrast to other areas of applications where datasets with tens or hundreds of thousands of variables are available forming a statistically significant population, data acquired by a rover driving over natural terrain present many challenges such as sparseness, presence of unknown and uncontrolled disturbances, dependence on the specific time and site of the acquisition.

The objectives pursued by feature selection include improvement in the prediction performance, reduction in training time, computational burden and memory usage of the algorithm, and facilitation of understanding the underlying process that generated the data.

The other contribution of this study is the adoption of a suitable learning algorithm to infer the type of terrain from the selected feature set. This algorithm will have to look for patterns in the data to construct the mapping from the proprioceptive measurements to the corresponding terrain type.

The well-known support vector machine (SVM) is contrasted with a deep convolutional neural network (CNN). While SVM requires in input hand-crafted features that are selected during a pre-processing stage, CNN uses learned features that are extracted automatically from the signal time histories.

An important goal of the proposed approach is to improve the performance of terrain classifiers for two use cases: generalization and extrapolation. Generalization is defined as the performance of an algorithm on previously unseen observations (test set) that is extracted from the same distribution as the data in the training set, for example, the same test run. The error measured on the test set corresponds to both the online performance of the model and the operating conditions included in the training set. The second use case, extrapolation, is even more challenging since, in general, learning algorithms are known to perform poorly outside the training data population. We compare the performance of the two terrain classifiers (SVM and CNN) for both generalization and extrapolation.

After related research is surveyed in Section 2, highlighting the novel aspects of this paper, Section 3 presents SherpaTT, the rover used as testbed, and the learning algorithms implemented in this study. Next, signal augmentation, feature extraction, and selection problems are tackled in Section 4. The results obtained from the terrain classifiers are presented and discussed in Section 5. Conclusions wrap up the paper.

2 | RELATED WORK

Solving terrain-related challenges such as soil identification is an important research area in autonomous robots, alongside trajectory planning, localization, and obstacle avoidance (Nampoothiri et al., 2021). The latest developments in terrain classification strategies show that researchers have been focusing on two main categories: visual (or exteroceptive) and visual-independent (or proprioceptive) methods. In both approaches, data collected from sensors are used to train machine or deep learning-based classifiers that enable identification of the traversed terrain.

The sensors used for visual perception include RGB cameras (Tai et al., 2017; Wellhausen et al., 2019), RGB-D cameras (Manduchi et al., 2005), LiDARs (Tai et al., 2017), visual cameras (Otsu et al., 2016) and monocular cameras (Barnes et al., 2017). Although visual-based approaches are more common than proprioceptive-based ones, they have limitations as well. The performance of RGB cameras

is limited by difficult environmental conditions (e.g., low, or direct lighting and surface reflectivity). LiDARs struggle to capture the fine texture of objects and terrains, and they also perform poorly in compromised environment conditions (e.g., in presence of dust, hail, and smog). Furthermore, vision-based rovers are not able to navigate in unfamiliar surroundings because observing distant terrain patches does not provide information about the mechanical properties that directly impact on vehicle mobility.

Therefore, researchers have investigated methods that use proprioceptive sensing for terrain classification. In this case, the sensors used to perceive the incoming terrain include inertial measurement unit (IMU), force-torque sensors, microphones, and wheel encoders. As an example, Ishikawa et al. (2021) used microphones to support an RGB camera in the dark conditions. Brooks and Iagnemma (2005) measured vibrations via accelerometers, analyzed them in the frequency domain and implemented an online classifier that relies on principal component analysis (PCA) for feature reduction. DuPont et al. (2008) presented a method based on frequency response and vibration-based transfer function. Giguire and Dudek (2011) used a tactile probe combined with accelerometers to account for inertial effects. Dutta and Dasgupta (2017) pursued a low-cost approach using a multisensor platform fitted with GPS, IMU, and metal detector. A model-based observer grounded in the Cubature Kalman filter was also proposed in Reina et al. (2020) to predict terrain deformability using vertical acceleration measurements.

The above works based on visual-independent approaches represent a step forward in the direction of providing a mobile robot with information about the mechanical properties of the terrain. Although they achieved high confidence levels, little effort was spent on feature selection as a means to reduce the computational burden of the model without penalties in performance. Ultimately, the objective of researchers that work on robot-terrain interaction is to develop an accurate algorithm that runs online while the robot is moving. This algorithm must comply with the limited resources of an autonomous vehicle in terms of processing power and memory. A reduction in the number of features used to train and test a machine learning classifier would lead to a lighter computational burden in terms of feature extraction time, testing time, and memory usage. One of the contributions of this paper is to develop a feature selection algorithm demonstrating that these benefits can be achieved without compromising the accuracy of the model.

A body of research has been devoted to the feature extraction process, as the quality of the feature space directly affects the accuracy of the associated classifier. The feature extraction strategy depends on the machine learning approach chosen for terrain classification. Traditionally, for a supervised machine learning algorithm such as SVM, an extraction stage is required where features are hand-crafted by experts based on their knowledge in the specific application domain. Attempts have been made in various research fields to find effective features, for example in image-processing-related applications (Lu & Weng, 2007). However, this approach is not always possible for classifiers and it is often practically difficult, for instance when the relationship between input measurements and user-defined classes is extremely complex or even completely unknown beforehand. Additionally, features that are crafted manually maybe not optimal. For this reason, finding more

systematic ways to get good features has drawn an increasing research interest (Bengio et al., 2013).

Notable progress has been done recently to find learning techniques that allow models to learn features automatically from data with minimal manual input. Solutions using deep neural networks (NNs) have especially attracted much attention. The effectiveness of deep NNs has been demonstrated in many fields other than image classification, such as audio and natural language processing or transfer learning. The adoption of recurrent and CNN was discussed in Vulpi et al. (2021), in the context of terrain classification using an agricultural robot equipped only with inertial and electrical current sensors. However, although the promising results, it remains challenging to evaluate the effectiveness of learned features contrasted with expert-designed ones. The complexity of this comparison resides in the difficulty of determining the descriptive power of hand-crafted features. For this reason, this paper presents a fair comparison between hand-crafted and learned features through a rigid feature scoring and selection process.

In previous work by the authors (Dimastrogiovanni et al., 2020), a preliminary attempt was presented to select a subset of optimal proprioceptive features to train an SVM-based ground classifier then tested over only two terrain types, for example, rock and sand.

In this study, several novel additions are made. First, a whole new signal engineering stage is introduced to improve the overall information content. The signal selection strategy is formalized and reflected in an explanatory block diagram. Improved robustness has been achieved by increasing the number of training repetitions for each candidate feature set. Then, the importance of feature selection for terrain classification is shown by comparing a machine learning approach (SVM) with a deep CNN in terms of model complexity, computational burden, and prediction accuracy over a larger terrain set (three types of terrain against two of the previous work). Finally, the system is evaluated not only in a standard generalization problem but as well as in two more challenging extrapolation contexts that are seldom described in the Literature.

3 | MATERIALS AND METHODS

The first part of this section (Section 3.1) briefly presents the experimental planetary rover used for data gathering, describing the onboard sensor suite and the data sets collected during the field trials for developing terrain classification models. Then, the learning algorithms for terrain classification are presented, providing insights into the theoretical background.

3.1 | The rover SherpaTT

The experimental test bed used in the ADE project is the SherpaTT rover (see Figure 1) built by DFKI (Cordes et al., 2018). SherpaTT is a hybrid four-wheeled-leg rover, where the wheel-on-leg design constitutes an actively articulated suspension system. Flexible metal wheels provide a passive ground adaption on a small scale, while the active suspension fits the wheel positions to larger ground irregularities (Cordes & Babu, 2016).

Each of the four legs of SherpaTT's suspension has five degrees of freedom (DOF): the rotation of the whole leg about the pan axis with respect to the robot body, the two rotations of the inner and outer leg parallelograms, the steer and drive angle of the wheel. A unique feature of Sherpa is a six-axis force-torque sensor (FTS) mounted on the flange of each wheel-drive actuator, providing direct measurement of the force system exchanged with the ground.

The rover also features a six-DOF manipulation arm. The arm is designed to withstand a good portion of the rover's weight to support it during locomotion. However, for the experiments described in this article, the arm was not involved in locomotion testing.

The logging system provides data at a rate of 100 Hz and comprises the following main proprioceptive blocks:

- IMU.
- Wheel-mounted six-axis load cell (LC). In this study, we adopt solely the LC mounted on the front left wheel.
- Joint telemetry (JT). Each of the 20 actuated joints of the suspension system delivers telemetry such as supply voltage, supply current, temperatures, PWM duty cycle, position (relative and absolute), and velocity.



FIGURE 1 SherpaTT in a sandy trench during the ADE final field tests in spring 2021 [Color figure can be viewed at wileyonlinelibrary.com]

The main data set used for this study was generated at the DFKI premises in Bremen, Germany. SherpaTT was remotely controlled to move for approximately 10 m in a straight line over three types of terrain: sand, gravel, and paved ground. This represents a varied data set with high traction, low deformability surface (paved ground) at one end, and a surface with low traction and high deformability (sand) on the other end, with gravel in the middle of the two (Figure 2). For each terrain, five runs were repeated in forward and reverse drive, except for gravel for which only four runs are available. Two different drive speeds of the rover were used, namely 0.1 and 0.15 m/s.

A second data set was generated in a sand mine close to Bremen (please refer again to Figure 1, GPS coordinates (DMS format): 53° 18' 54.9" N, 8° 41' 17.3" E) during the ADE's final testing in April 2021. This independent data set is used to predict terrain labels for observations outside the training data population. In this last environment, the surface traversed was somewhat like the sand case of the previous settings but the terrain was more compact and wetter. It can be directly observed in Figure 1 how humid sand got matted to the wheels while traversing, unlike in the previous environments (Figure 2).

3.2 | Learning algorithms

3.2.1 | Support vector machine

SVM is a well-established machine learning solution for soil classification problems (Bellone et al., 2018; Gonzalez et al., 2019; Reina et al., 2017). This section will present a summary of the theory behind SVM classification. For a detailed description of SVM algorithm please refer to Hastie et al. (2009) and Vapnik (2013). An SVM problem is composed of two stages: training and testing. Given two classes A and B (binary classifier), an input training set S composed of p samples and n features can be defined as:

$$S = \{(x_i, y_i) : x_i \in \mathbb{R}^n, y_i \in \{-1, 1\}, \\ i = 1, 2, \dots, p\}, \text{ where } \begin{cases} y_i = 1 & \text{if } x_i \in A \\ y_i = -1 & \text{if } x_i \in B \end{cases} \quad (1)$$

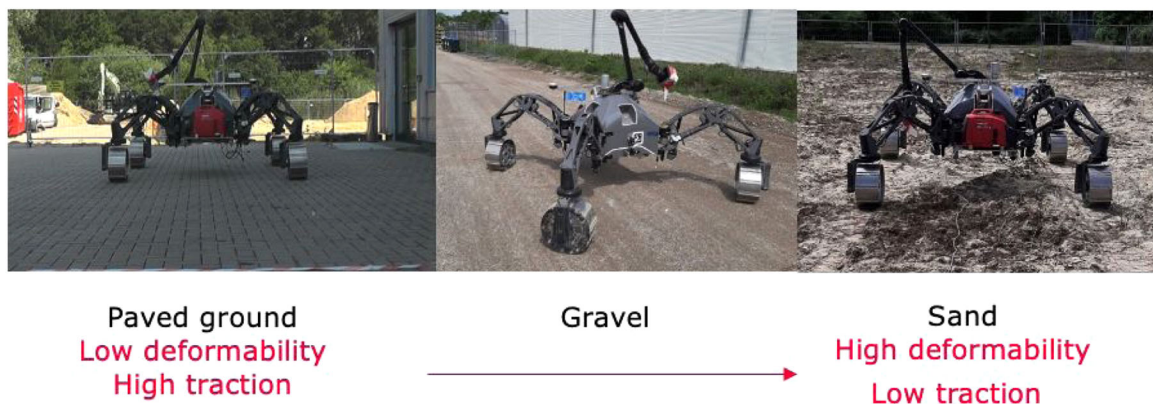


FIGURE 2 Types of surfaces traversed by SherpaTT during the test and development of the system [Color figure can be viewed at wileyonlinelibrary.com]

x_i are referred as predictors, y_i represents the response variable. The purpose of the linear SVM algorithm is to find a decision function D that allows, in the testing phase, to classify any new sample $x \in \mathbb{R}^n$ according to the sign of $D(x)$. This is done by finding the hyperplane that maximizes its distance to the support vectors (i.e., the predictors closest to the hyperplane), while minimizing the loss due to misclassification. The Lagrangian dual of this optimization problem can be formulated as follows:

$$\begin{aligned} \max_{\alpha} \quad & \sum_{i=1}^p \alpha_i - \sum_{i=1}^p \sum_{j=1}^p \alpha_i \alpha_j y_i y_j x_i^T x_j \\ \text{subject to} \quad & \sum_{i=1}^l y_i \alpha_i = 0 \quad \text{and} \quad 0 \leq \alpha_i \leq C, \end{aligned} \quad (2)$$

where α_i are Lagrangian multipliers and C is a parameter called box constraint.

The dominant approach for multiclass applications is to reduce the single problem into multiple binary classification problems (Duan & Keerthi, 2005). One of the most common methods for such reduction is the error-correcting output codes (ECOC) model (Dietterich & Bakiri, 1994). The most important parameter for this method is the coding design, a matrix where elements indicate which classes are trained by each binary learner, reducing the multiclass problem to a series of binary problems.

In this study, SVM is considered as the benchmark approach that is compared against other alternatives as a deep CNN.

3.2.2 | Convolutional neural network

In contrast to SVM that uses handcrafted features manually engineered by data analysts, CNN derives features automatically from inputs throughout a training process, searching for those that better characterize each terrain. However, as input, CNN takes an image-like observation, therefore a first practical issue to solve is how to derive a 3D object from several signals. One possible solution, proposed in this study, is to resort to fast Fourier transform (FFT) to construct magnitude spectrograms of the signals then appended into

a multichannel object forming the input for the net. So, sensory data can be assembled in 3D shape, namely height, width and depth. The height corresponds to the frequencies (nF) analyzed by the FFT, the width corresponds to the number of time windows (nW) adopted in the spectrogram, and the depth is the number of signals (nCh).

The architecture of the CNN is shown in Figure 3, where the neural dimensions and the learnable variables of each layer are indicated.

The first layer takes as input the multichannel spectrogram, next, the batch-normalization layer normalizes inside the mini-batch the value kept by each input neuron. Normalization process follows Equation (3) where x_n and y_n are respectively, the input and output values of neuron n of this layer, batch mean μ_B and SD σ_B are computed during training, while learnable parameters offset γ and bias β are searched through optimization across the whole training set. Computational constant ε can improve numerical stability when variance σ_B^2 is small.

$$\begin{aligned} y_n &= \gamma \frac{x_n - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}} + \beta \\ \forall n &= 1 \dots (nW \cdot nF \cdot nCh) \end{aligned} \quad (3)$$

The following 2D convolution layer spans the output across time and frequency domain convoluting the $nW \times nF \times nCh$ batch-normalized spectrograms into $nFilt$ objects of dimensions $nW \times nF$. A user-specified number of square filters $nFilt$ with size fsz are here used to perform convolution process briefly described in Equation (4), where X is the zero-padded neural grid after batch-normalization and Y the output of the convolution process. The learnable parameters of this layer are the kernel of filter m , the weights of matrix $_m\omega$, and $_m\beta$ the corresponding bias.

$$\begin{aligned} Y_{w,f,m} &= \sum_{c=1}^{nCh} \sum_{i,j=-fsz/2}^{fsz/2} m\omega_{i,j,c} * X_{w+i,f+j,c} + m\beta \\ \forall w &= 1 \dots nW, \forall f = 1 \dots nF, \forall m = 1 \dots nFilt \end{aligned} \quad (4)$$

The output of the convolution process is passed to the Rectified Linear Unit (ReLU) activated neurons in a grid $nW \times nF \times nFilt$, fully connected to nCl neurons where nCl is the number of terrain classes considered. Compared to other activation functions such as the

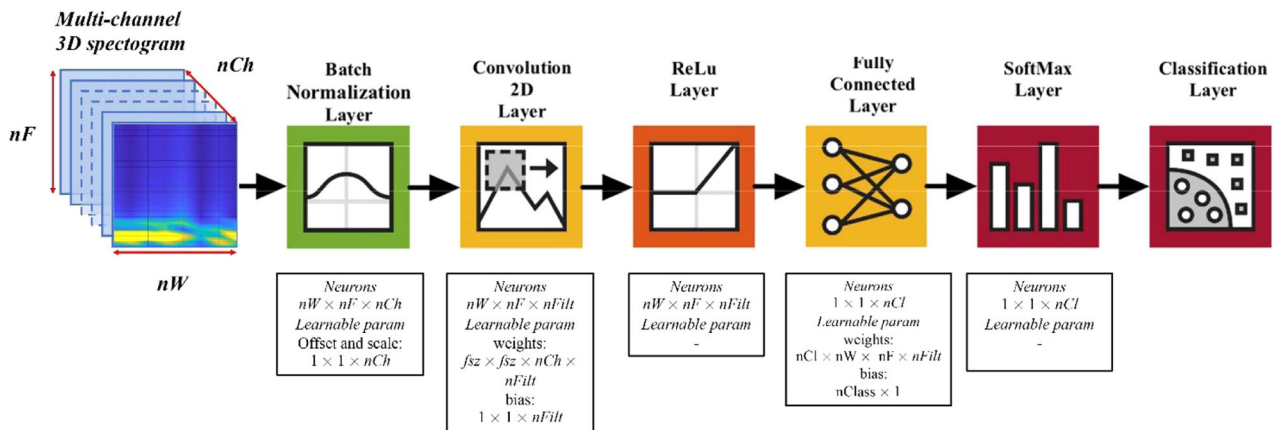


FIGURE 3 Architecture of the convolutional neural network [Color figure can be viewed at wileyonlinelibrary.com]

sigmoidal function, ReLu helps in preventing the exponential growth in the neural network computation and the “vanishing gradient” problem that is the tendency for the gradient of a neuron to approach zero for high values of the input (Kingma & Ba, 2015). The two following layers SoftMax and Classification are standard as output layers for classification networks. The function SoftMax is defined in Equation (5) where x_n is the n th input neuron and y_n is the corresponding output of this layer.

$$y_n = \frac{\exp(x_n)}{\sum_{i=1}^{nCI} \exp(x_i)}, \forall n = 1 \dots nCI \quad (5)$$

The output layer of this network is the classification layer that computes the cross-entropy loss for classification among terrains.

3.3 | Parameters of the learning algorithms

In this section, the values assigned to the parameters of the learning algorithms are highlighted.

The parameter set of the SVM-based classifier is indicated in Table 1. It was found empirically to give the best balance of sensitivity and specificity (Lin et al., 2002).

As for CNN, during the training stage the learnable parameters are updated at each iteration, whereas the hyper-parameters are defined by the user to govern the training process.

In one iteration, the network analyses the samples contained in the mini-batch. One epoch consists in the number of iterations necessary to review the entire training data set. The training stage stops after the network has passed through the entire data set the number of times specified as the maximum number of epochs. It is usually preferred to stop the training before this number has been reached, not only because it shortens the time required for training, but also because it prevents overfitting on the training set. Therefore, a percentage of the training data is kept apart as a validation set, and the network evaluates its loss after the number of iterations specified as validation frequency. The validation patience is the number of times that this loss can be smaller or equal to the previously smallest loss before the training stage stops. The initial learning rate drops by a factor (learn drop factor) after a given number of iterations (learn drop period). Part of the hyperparameters is set according to the Literature, for example, the solver and the gradient threshold follow the value suggested in Kingma and Ba (2015). The remaining parameters have been selected empirically through grid-search and they are reported in Table 2.

Note that for a fair comparison with SVM, the magnitude spectrograms of the signals used as input to CNN are obtained from a time window $w_s = 2$ s (please refer to Section 4.2).

TABLE 1 Parameters of the SVM classifier

Parameter	Value
C (Box constraint)	1
Standardize	True
Coding design	One-versus-one

4 | SIGNAL ENGINEERING

A list of measurements available from the SherpaTT's sensor suite is shown in Table 3, with corresponding sensorial group and Signal ID. From the first analysis of Table 3, some of the signals may appear seemingly correlated. However, if we consider, for example, body acceleration and wheel force, these signals are actually uncorrelated through the flexibility of the suspension system, and therefore they are both relevant for the proposed analysis.

Signals that are directly derived from measurements are referred to as direct signals. Conversely, signals engineered with expert knowledge combining direct signals are referred to as indirect, as explained in the next section.

Figure 4 shows a sample time history of the vertical acceleration (gravity-compensated) and drive torque experienced by SherpaTT on different terrains. As seen from this figure, signals show a signature that seems to change according to the specific surface. The goal of this study is to learn this signature to gain terrain awareness. To this

TABLE 2 Hyper-parameters of the CNN classifier

Parameter	Value
Filter size (fsz)	[5, 5]
Number of filters (nFilt)	9
Mini-batch size	160
Maximum number of epochs	150
Validation percentage	15%
Validation frequency	20
Validation patience	15
Initial learning rate	0.005
Learn drop factor	0.2
Learn drop period	10

TABLE 3 List of available proprioceptive signals

Signal	Symbol	Sensors	Signal ID
Longitudinal force	F_x	LC	S1
Vertical force	F_z	LC	S2
Drive torque	T_d	LC	S3
Drive electrical current	C_d	JT	S4
Drive PWM duty cycle	PWM_d	JT	S5
Longitudinal acceleration	a_x	IMU	S6
Lateral acceleration	a_y	IMU	S7
Vertical acceleration	a_z	IMU	S8
Gyro roll rate	$gyro_x$	IMU	S9
Gyro pitch rate	$gyro_y$	IMU	S10
Gyro yaw rate	$gyro_z$	IMU	S11

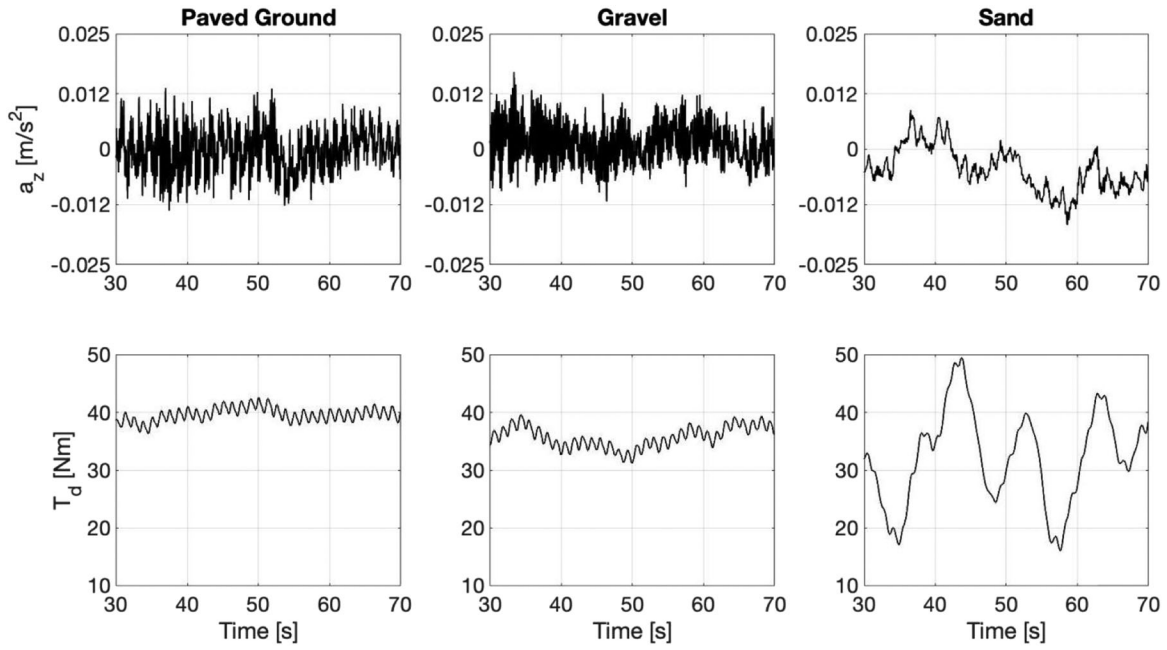


FIGURE 4 Vertical acceleration and drive torque (wheel front left) measured while SherpaTT driving straight on different terrains

aim, it is necessary to select the most relevant signals for building an accurate predictor.

4.1 | Signal augmentation

To improve the information content, an augmentation engine combines multiple direct measurements based on our understanding of the physical mechanisms underlying the wheel–terrain interaction. These are a few of the many possible signal combinations that can be implemented, and they are chosen following a trial-and-error approach to provide the best performance over other alternatives. In this way, nine more indirect signals can be obtained (Table 4). The derivation of these signals is detailed in this section, and the rationale behind the choice of these entities is also explained.

Two main motivations support the proposed augmentation stage. First, two or more signals that are useless (not relevant) for themselves can be useful when combined. Then, noise reduction and consequently better class separation may be achieved by adding variables that are seemingly redundant (Guyon & Elisseeff, 2003). This explains why we resort to indirect or combined signals and include redundant measurements of the same physical quantity.

The first indirect signal is the power loss due to the wheel traction on a given terrain. It can be derived from a “mechanical” or “electrical” analysis. The mechanical power can be estimated as follows:

$$P_M = T_d \cdot \omega, \quad (6)$$

where ω is the rotational speed of the wheel. Conversely, the electrical power consumption can be obtained as follows:

TABLE 4 List of indirect signals

Signal	Symbol	Sensors	Signal ID
Mechanical power	P_M	LC, JT	S12
Electrical power	P_E	MC	S13
Vertical force offset	dx	LC	S14
Friction coefficient 1	μ_1	LC	S15
Friction coefficient 2	μ_2	LC	S16
Friction coefficient 3	μ_3	LC, JT	S17
Speed deviation	SD	JT	S18
Normalised speed deviation	SD_n	JT	S19
Sinkage	z	LC	S20

$$P_E = \eta \cdot V_d \cdot PWM_d \cdot C_d, \quad (7)$$

where V_d is the drive voltage, C_d is the wheel drive current, PWM_d is the duty cycle of the wheel drive Pulse Width Modulation, and η is the efficiency of the electric motor, assumed to be constant and approximately equal to 0.85.

Due to the rolling resistance, the direction of the resultant vertical force F_z might not pass through the centre of the wheel, with an offset in the direction of the movement (Figure 5). This is especially true for soft terrain where the impact of rolling resistance is larger.

Therefore, we can define the vertical force offset dx from the equilibrium of moments around the centre of the wheel, neglecting the contribution of rotational inertia:

$$dx = \frac{T_d - F_x \cdot R}{F_z} \quad (8)$$

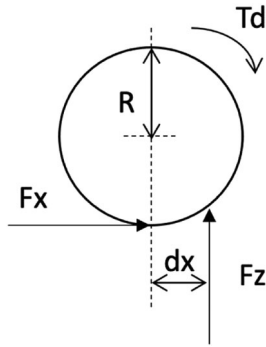


FIGURE 5 Definition of vertical force offset (dx)

where R is the loaded wheel radius defined as:

$$R = R_N - \frac{F_z}{k_z} \quad (9)$$

being R_N ($=200$ mm) the nominal wheel radius, and k_z the vertical stiffness of the SherpaTT wheel that was experimentally estimated as 69 N/mm.

The friction coefficient is an important entity related with the traction ability over the traversed surface. In this study, it is estimated in three different ways:

$$\mu_1 = \frac{F_x}{F_z} \mu_2 = \frac{T_d}{F_z \cdot R} \mu_3 = \frac{C_d k_T}{F_z \cdot R}, \quad (10)$$

where k_T (17.4 Nm/A in our case) is the scale factor taking into account the torque constant of the electric motor and the transmission ratio of the motor reducer.

Speed deviation is the difference between the angular speed of each wheel ω and the average angular speed of the four wheels $\bar{\omega}$. In this study, speed deviation was estimated in two ways:

$$SD = |\omega - \bar{\omega}| \quad SD_{\text{normalised}} = \frac{\omega - \bar{\omega}}{\bar{\omega}}. \quad (11)$$

Wheel sinkage is another critical parameter related to rough terrain mobility that can be approximated as suggested in Guo et al. (2020):

$$z = R \cdot \left(1 - \cos \left(2 \cdot \frac{dx}{R} \right) \right). \quad (12)$$

One important aspect is the general data consistency. As an example, Figure 6 shows the drive torque delivered by the left-wheel drive motor, measured by three different sensors. Direct torque measurement from the wheel-mounted LC is denoted by a solid grey line, whereas indirect estimation via the associated electric current drawn by the motor is marked by a black solid line. Finally, an alternative indirect measurement via the LC-derived longitudinal force is also plotted using a dashed black line. As seen in this figure, all three measurements show a good agreement. Similar results were observed on different surfaces.

4.2 | Feature extraction

First, each sensory signal is divided in time windows, and then, for each window features are extracted as the four main statistical moments. The size of the window, w_s is a design parameter. It is set as $w_s = 2$ s corresponding to a traversed terrain patch of about 20 cm (comparable with the wheel radius) at an average travel speed of 0.1 m/s. In previous works by the authors (Vulpi et al., 2021), it was found that this value of window size represents a good trade-off between informative content and spatial resolution.

The four statistical moments are mean E , variance σ , skewness Sk and kurtosis Ku and are defined as follows:

$$E_i = \frac{1}{N} \sum_{n=1}^N x_n \quad \sigma_i^2 = \frac{1}{N} \sum_{n=1}^N (x_n - E_i)^2 \quad Sk_i = \frac{1}{N} \frac{\sum_{n=1}^N (x_n - E_i)^3}{(\sqrt{\sigma_i^2})^3} \quad Ku_i = \frac{1}{N} \frac{\sum_{n=1}^N (x_n - E_i)^4}{(\sqrt{\sigma_i^2})^4} \quad (13)$$

where x_n is the value of the signal at the n th time step and N is the total number of time-steps for the i th window.

The extraction of the statistical features brings the size of the SVM-feature space to 80 (20 signals multiplied by their four statistical moments). The generic feature will be indicated as $SiMj$, where i ($i = 1, \dots, 20$) represents the signal ID, whereas j represents the statistical moment ($j = 1, \dots, 4$).

4.3 | Feature selection

Retaining only the features with the highest information content reduces the computational cost while preserving the accuracy of the model. The selection process can be performed via feature scoring using appropriate validity indices. Then, an iterative search algorithm can be followed to select a reduced best feature space.

4.3.1 | Validity indexing

A validity index can be assigned to each feature. This index represents a measure of the information content of the feature. In this study, two validity indices are considered: the Pearson Coefficient (PC) (Hastie et al., 2009), and the WB index (Zhao & Fränti, 2014).

The PC index can be computed through linear regression of a feature against the 3 classes of terrain, for example, sand, gravel, and paved ground. The higher the PC, the larger the information content of the feature. Although this index can be successfully used for 2-class classification problems (Dimastrogiovanni et al., 2020), it might be difficult to implement it for multiclass cases like the one presented in this study, because the number assigned to each type of terrain is arbitrary. To overcome this issue, first, the PC index is computed for each terrain pair (e.g., sand-gravel, gravel-paved ground, and sand-paved ground), and then averaged. For example,

FIGURE 6 Torque applied by the left front wheel of SherpaTT as obtained from direct measurement of the load cell (solid grey line), indirect measurement via the electric current drawn by the motor (solid black line), or alternatively via the longitudinal force provided by the load cell (dashed black line)

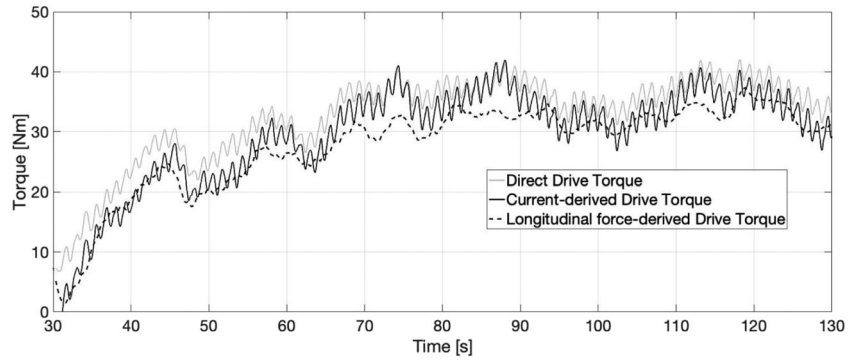
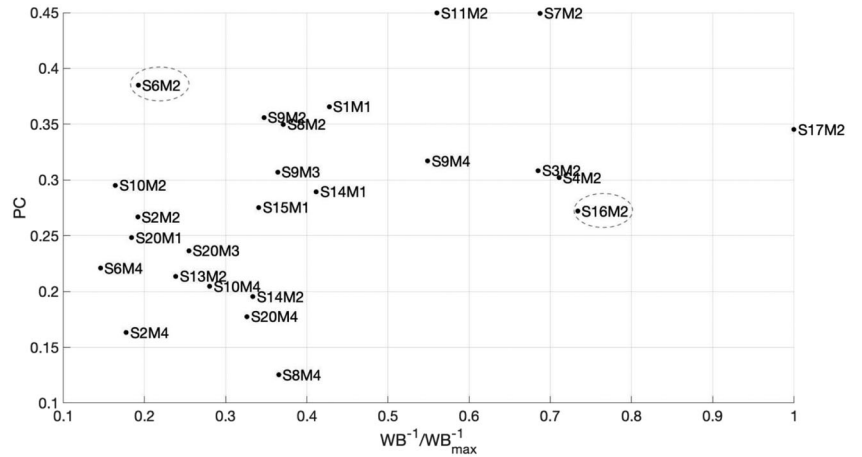


FIGURE 7 PC and WB indices distribution for the most relevant features



the PC index of the feature $SiMj$ against the Classes 1 and 2 (sand and gravel) can be calculated as (Guyon & Elisseeff, 2003):

$${}_2^1PC_{SiMj} = \frac{\text{cov}({}_2^1F_{SiMj}, {}_2^1y)}{\sqrt{\text{var}({}_2^1F_{SiMj})\text{var}({}_2^1y)}} \quad (14)$$

where ${}_2^1F_{SiMj}$ is a vector containing all values of the feature $SiMj$ for terrains 1 and 2, whereas ${}_2^1y$ contains class values (1 or 2) for each element of ${}_2^1F_{SiMj}$. Similarly, ${}_3^2PC_{SiMj}$ (PC index of feature $SiMj$ against the classes gravel and paved ground) and ${}_3^1PC_{SiMj}$ (PC index of feature $SiMj$ against the classes sand and paved ground) follow the same principle.

The overall PC index for feature $SiMj$ can be now computed as follows:

$$PC_{SiMj} = \frac{{}_2^1PC_{SiMj} + {}_3^2PC_{SiMj} + {}_3^1PC_{SiMj}}{3} \quad (15)$$

In addition, the WB index can be computed for feature $SiMj$:

$$WB_{SiMj} = m \cdot \frac{SSW_{SiMj}}{SSB_{SiMj}} \quad (16)$$

where SSW is the sum of a square within classes and SSB is the sum of squares between classes, computed as follows:

$$SSW_{SiMj} = \sum_{k=1}^{nCl} \sum_{s=1}^{n_k} (x_s - \mu_k)^2, \quad SSB_{SiMj} = \sum_{k=1}^{nCl} n_k (\mu_k - \mu)^2 \quad (17)$$

where x_s is the s th sample of feature $SiMj$, μ_k is the class k centroid value, μ is the overall data set centroid value, n_k is the number of samples in class k and nCl ($=3$) is the number of classes. A low value of WB_{SiMj} indicates that classes form compact and distant clusters relatively to feature $SiMj$. Therefore, the score assigned to each feature will be WB^{-1} : the higher the WB^{-1} , the better the feature for classification purposes.

The rationale behind using two validity indices is that the WB and PC have two different statistical meanings: the former describes the compactness of classes, the latter shows the correlation between a given feature and the type of terrain. One may think that a feature with a low value of PC index will also have a relatively low value of WB^{-1} index. However, this is not always true, and exceptions do occur. For example, Figure 7 shows the distribution of PC and WB indices for the 25 features with the highest scores. S6M2 is the feature with the third highest value of PC index, but it is only the 21st feature in terms of WB^{-1} . Similarly, S16M2 is the feature with the second-highest value of WB^{-1} index, but it is only the 14th in terms of PC. This shows that the two indices rank the features in different ways, therefore they complement each other very well.

4.3.2 | Selection algorithm

The proposed selection approach is based on the iterative search scheme presented in the block diagram of Figure 8. The input to the algorithm is the full set of n_{feat} ($=80$) features. These features are then ranked using

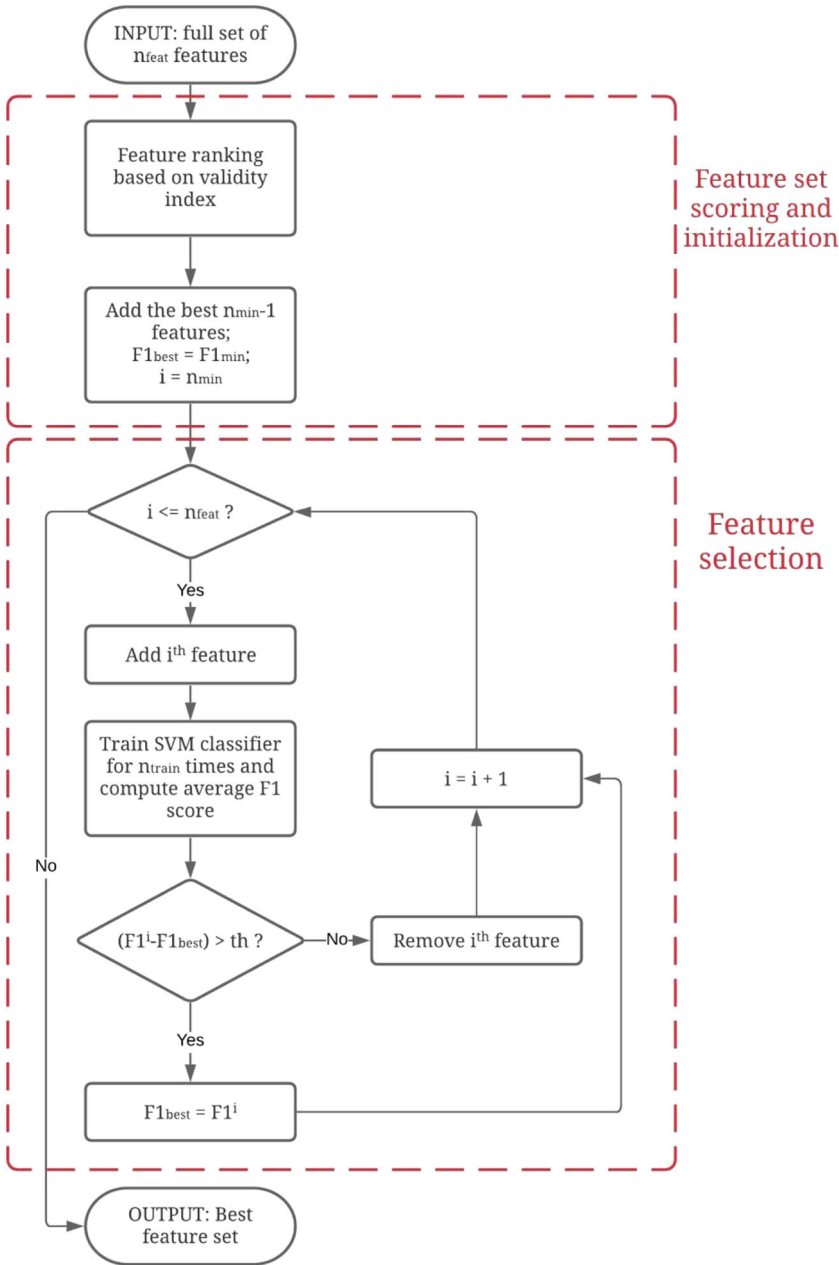


FIGURE 8 Block diagram of the proposed feature selection algorithm [Color figure can be viewed at wileyonlinelibrary.com]

the output of one of the two validity indices (PC or WB) as a score. The best feature set is initialized with the first $n_{\min}-1$ ($=2$) features of the ranking. At this point, the objective is to iterate on all the remaining features to find those which provide better classification performance. In each iteration, identified with the index i that varies from n_{\min} to n_{feat} , the i^{th} feature in the ranking is added provisionally to the best feature set. Then, an SVM-based classifier is trained and evaluated in terms of F1 score via fivefold cross-validation. The k -fold cross-validation process partitions data into k randomly chosen subsets (or folds) of roughly equal size. Therefore, to improve the robustness of the feature selection algorithm, the training phase is repeated n_{train} ($=10$) times and the final F1 score is computed as the average of the scores obtained at each training phase. If the final F1 score is sufficiently higher than the best F1 score obtained so far, the i^{th} feature is kept in the best feature set, and the best

F1 score is updated. Otherwise, the i^{th} feature is discarded from the best set and not considered for training purposes.

To facilitate the reading of the block diagram in Figure 8, the meaning and the numerical values of the parameters involved in the selection process are collected in Table 5.

The selection process discussed in Figure 8 can be repeated for each one of the two validity indices. Eventually, two best reduced feature spaces will be obtained: one associated with the PC and the other with the WB index. To further improve the robustness of the selection algorithm, the union of these two sets is chosen as the best for SVM training purposes. The 18 selected features are listed in Table 6. It is worth noting that three features extracted from indirect signals are included as well, thus, proving the utility of the signal augmentation phase.

TABLE 5 List of parameters involved in the feature selection approach

Parameter	Description	Value
n_{\min}	Minimum number of features	3
$F1_{\min}$	Minimum F1 score	60%
th	Accepted improvement (threshold) in the F1 score	5%
n_{train}	Number of trainings for each new best feature set	5
n_{feat}	Number of features in the initial full feature set	80

TABLE 6 Best feature set

Signal	Statistical moment	Direct or Indirect	Feature ID	$WB^{-1}/WB^{-1}_{\text{MAX}}$	PC
μ_3	Variance	Indirect	S17M2	1.00	0.345
a_y	Variance	Direct	S7M2	0.691	0.449
T_d	Variance	Direct	S3M2	0.685	0.308
$gyro_z$	Variance	Direct	S11M2	0.561	0.450
$gyro_x$	Kurtosis	Direct	S9M4	0.549	0.317
F_x	Mean	Direct	S1M1	0.428	0.366
a_z	Variance	Direct	S8M2	0.371	0.350
$gyro_x$	Skewness	Direct	S9M3	0.364	0.307
$gyro_x$	Variance	Direct	S9M2	0.348	0.356
μ_1	Mean	Indirect	S15M1	0.342	0.275
z	Kurtosis	Indirect	S20M4	0.327	0.177
a_x	Variance	Direct	S6M2	0.192	0.385
F_z	Variance	Direct	S2M2	0.192	0.267
$gyro_y$	Variance	Direct	S10M2	0.164	0.295
F_x	Variance	Direct	S1M2	0.143	0.253
a_y	Kurtosis	Direct	S7M4	0.048	0.115
PWM_d	Mean	Direct	S5M1	0.013	0.062
$gyro_z$	Mean	Direct	S11M1	0.011	0.064

A 3D plot of the three most relevant features in terms of WB index is shown in Figure 9 to help the reader to easily visualize the result of the whole selection process. As shown in this figure, the sand data form a quite compact cluster, with relatively low values of all three features. Conversely, gravel and paved ground data show higher values of S7M2 (variance of a_y) than sand and differentiate prevalently for values of S17M2 (variance of μ_3).

5 | RESULTS AND DISCUSSION

In this section, the results of the generalization problem are shown on the main data set. Next, results for two extrapolation cases are presented.

5.1 | Generalization

In the generalization problem, only the main data set is used (e.g., experiments on paved ground, gravel, and sand). The algorithms are tested via fivefold cross-validation. The data set comprises of 1204 samples, where a sample corresponds to a 2-second time window. Of these 1204 samples, 443 are collected on paved ground, 338 on gravel, and 423 on sand.

One of the objectives of this paper is to demonstrate how a proper feature selection algorithm can reduce the computational and memory cost of the model, while maintaining a similar accuracy in prediction. Table 7 shows a comparison between the two machine learning algorithms in terms of accuracy and computational burden. Moreover, SVM is tested with three different feature sets:

- Direct feature set (44 features).
- Full feature set (80 features).
- Best feature set (18 features).

while CNN is tested with three different signal sets:

- Direct signal set (11 signals).
- Full signal set (20 signals).
- Best signal set (13 signals).

The signals used for training CNN correspond to those used to compute SVM features. In fact, the 44 direct features are the four statistical moments of the 11 direct signals, and the full 80-feature set is composed of the four statistical moments of the full 20-signal set. Furthermore, the training set for CNN includes the signals used to derive the features in the best feature set. Namely, the 13 best signals are: friction coefficients 1 and 3, longitudinal, lateral, and vertical accelerations, drive torque, yaw, pitch and roll rates, longitudinal and vertical forces, sinkage, drive PWM.

The accuracy of the SVM model trained with the direct and full feature sets is 89.8% and 90.8%, respectively. With the full feature set, more samples are correctly classified by SVM, but memory usage has increased by 82%, training time by 32%, testing time by 71%, and feature extraction time by 50%. This proves the effectiveness of the signal augmentation in terms of accuracy and shows the drawbacks in terms of the computational burden. The purpose of feature selection is to reduce the computational cost, without losing classification accuracy. The results presented for SVM trained with the best feature set, prove that the feature selection algorithm proposed in this study is effective. In fact, the accuracy reaches 90.9% and when compared to the SVM trained on the full feature set, while the model memory usage is reduced by 77%, training time by 6%, testing time by 29%, feature extraction time by 33%.

The effectiveness of both input signal augmentation and feature selection is also confirmed by the results presented for CNN. This deep learning algorithm gains in terms of accuracy from signal augmentation reaching 96.4%. Using the full signal set still results for CNN in the same drawbacks presented for SVM: model memory usage increased by 18%, training time by 36%, feature extraction time by 77%. In contrast with

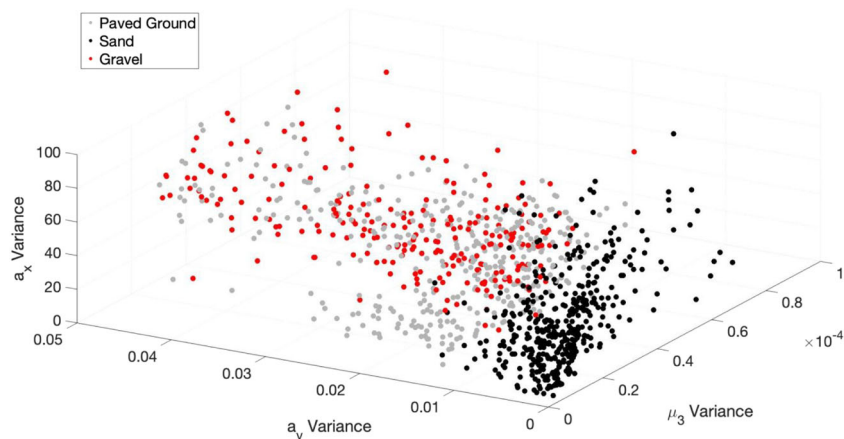


FIGURE 9 3D plot of the first three features with the highest score of WB^{-1} [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 7 Performance comparison between terrain classifiers trained on different feature sets: direct, full, best feature set

Feature and signal sets	SVM			CNN		
	Direct	Full	Best	Direct	Full	Best
Accuracy (%)	89.8	90.8	90.9	95.6	96.4	96.2
Model memory usage (kB)	547.6	996.9	228.0	44.9	53.2	45.8
Training time (ms)	118.9	157.7	148.0	1.07 e4	1.46 e4	1.06 e4
Testing time (ms)	17.4	29.8	21.2	153.0	119.4	114.7
Feature extraction time (ms)	0.6	0.9	0.6	2.6	4.6	2.7

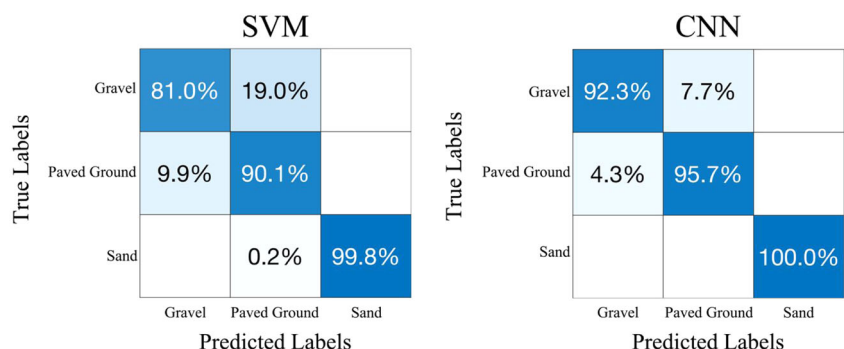


FIGURE 10 Generalization results for best features SVM and best signals CNN. CNN, convolutional neural network; SVM, support vector machine [Color figure can be viewed at wileyonlinelibrary.com]

SVM, testing time for CNN with full signal set is reduced by 22%. Training CNN with the best signals resulting from feature selection leads to an accuracy of 96.2% and when compared to the full-signal CNN, the model memory usage is reduced by 14%, training time by 27%, testing time by 4%, feature extraction time by 41%.

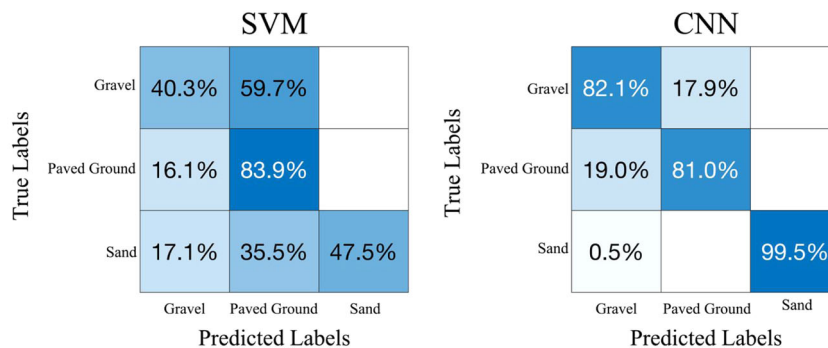
Feature extraction times presented in the last row of Table 7 are suitable for online applications for both SVM and CNN, even if construction of multichannel spectrograms from best signals for CNN takes about 2.1 ms more than the construction of best features for SVM. It should also be noted that feature extraction time for both SVM and CNN can be further improved by optimizing the current MatLab code using vectorization or processing the data directly with a C++ code. Note that at the time of writing of the paper, the algorithms and the data set are under revision in a private Github repository that will be made available to the interested readers upon paper publication.

TABLE 8 Accuracy, Precision, Recall and F1 score for SVM and CNN in generalization

Class	SVM			CNN		
	Gravel	Paved ground	Sand	Gravel	Paved ground	Sand
Precision (%)	89.1	82.4	100	80.3	82.2	100
Recall (%)	81.0	90.1	99.8	92.3	95.7	100
F1 score (%)	84.9	86.1	99.9	85.9	88.4	100

Confusion matrixes for both SVM and CNN are shown in Figure 10 only for best feature and best signal sets. Sensitivity results for each class are contained in the diagonal elements of each confusion matrix. The performance of both models in terms of precision, recall, and F1 score are shown in Table 8. Both models perform good in the generalization of data, with CNN being slower but significantly

FIGURE 11 Extrapolation results for best features SVM and corresponding signals CNN. CNN, convolutional neural network; SVM, support vector machine [Color figure can be viewed at wileyonlinelibrary.com]



more accurate. This increase in classification accuracy is not the main advantage for CNN classification model with respect to SVM. Where the two models show the greatest difference in classification performance is indeed extrapolation, as shown in the next section.

5.2 | Extrapolation

In the extrapolation problem, the operating conditions of training and testing sets are different, therefore these sets do not come from the same population. In this study two extrapolation cases are presented. The first one deals with varying rover speed, whereas the second one assesses the performance of the algorithms on a terrain unseen in the training phase.

5.2.1 | Testing on a new vehicle velocity

During the experiments with SherpaTT, the rover was controlled at two different speeds: 0.1 m/s and 0.15 m/s. Of the 14 runs, 7 were conducted at low speed (0.1 m/s) and 7 at high speed (0.15 m/s). Data collected at low-speed form the low-speed distribution, whereas data collected at high-speed belong to the high-speed distribution. In the extrapolation problem presented here, low-speed data are used as training set, while high-speed data are used as testing set. Both sets belong to the main data set (paved ground, gravel, and sand).

Proprioceptive sensorial data are very useful for terrain classification but also show a strong dependency from traversing speed (Bai et al., 2019). Most terrain classification algorithms analyse and classify proprioceptive data acquired at constant traversing velocity on different terrains. Studies have been also conducted to show the dependency of terrain classification performances from rover's traversing speed, searching for the velocity that maximizes classification performance. For being able to classify the traversed terrain at any travelling speed a rover should be equipped with a model trained on a vast variety of possible traversing speeds or could only use speed-independent features that are difficult to construct and may not be well suited for terrain classification. Another way of achieving the goal of sensing and classifying the terrain at any travelling speed is using a model that shows good results when tested on data acquired

TABLE 9 Precision, Recall and F1 score for SVM and CNN in extrapolation using varying velocity

Class	SVM			CNN		
	Gravel	Paved ground	Sand	Gravel	Paved ground	Sand
Precision (%)	54.8	46.8	100	80.3	82.2	100
Recall (%)	40.3	83.9	47.5	82.1	81.0	99.5
F1 score (%)	46.4	60.1	64.4	81.2	81.6	99.7

at a traversing velocity different from the one used for training. Figure 11 contains the confusion matrixes for both SVM and CNN when trained on low-speed data and tested on high-speed ones. As can be seen, despite both models showed good results in generalization only CNN is also capable of extrapolating the information of the traversed terrain from data acquired at a different speed. The two models were still trained and tested using only best feature set for SVM and corresponding signal set for CNN. While CNN keeps classification accuracy as high as 89.5%, SVM becomes unreliable achieving only 55.7% of correctly classified data samples. The performances of both models in terms of precision, recall, and F1 score are shown in Table 9.

It should also be pointed out that high-speed data used as testing constitute 50% of available data, representing, therefore, testing set larger than the one usually used (20%–30%). The robustness of CNN's classification performance on a large testing set composed of data acquired at a different speed suggests that this model is well suited for terrain classification purposes. Moreover, the features automatically learned from signal spectrograms appear to be more reliable than statistic ones and represent a better choice to be able to classify the traversed terrain at various traveling velocities.

Similar results are obtained when trained on high-speed data and tested on low-speed data, and they are omitted for brevity sake.

5.2.2 | Testing on an independent data set

The second extrapolation use case aims to evaluate the system response when labeling observations collected on a terrain different from those used in training (independent data set). To this aim, the

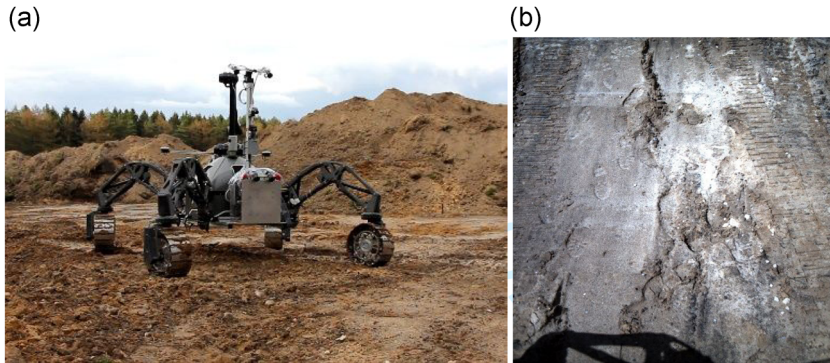


FIGURE 12 (a) Sherpa TT during the sand mine testing; (b) a close up of the tracks left by the wheels [Color figure can be viewed at wileyonlinelibrary.com]

ground classifier previously trained on the main data set (formed by paved ground, gravel, and sand) is further validated on a representative data set gathered from a second field test campaign run in a planetary analogue terrain in a sand mine near Bremen (see Figure 12).

For this extrapolation challenge, we have tried to generalize the classification problem at hand by referring to terrain difficulty labels rather than specific terrain classes, as explained in Table 10. Adopting the proposed terrain difficulty scale, paved ground and sand can be seen as the opposite extremes. Firm-ground offers better traction and less compressibility, therefore a low-difficulty label can be assigned to it. Conversely, soft ground poses more challenges, and it is scored as a highly difficult surface. Then, the difficulty degree associated with an unknown observation can be considered as inversely proportional to the distance from the class sand. One should note that such a generalization effort can be useful or necessary for the practical implementation of planetary exploration terrain classifiers that can be only trained on Earth using representative analogue surfaces, and then applied to unknown planetary surfaces via extrapolation.

The sand mine independent data set consists of 302 samples, where, again, a sample corresponds to a 2-second window. It should be also underlined that, although ground-truth data is not available for this extrapolation problem, the terrain in the sand mine can be expected as a surface with medium-high difficulties, like the sand type of the main data set (Figure 2) but somewhat more compact and humid. As an indicative measure, sample tracks left by the wheels on the sand mine terrain are shown in Figure 12b.

The classification results obtained from SVM and CNN are collected in Table 11 showing predicted labels of terrain difficulty. Out of the 302 samples, the SVM-based algorithm classifies 71.2% as high difficult terrain, 17.2% as medium and 11.6% as low. CNN performs similarly, classifying 69.9% of the new terrain samples as highly difficult, 24.2% as medium, and 5.9% as low. A relatively low percentage of the test samples (about 12% for SVM and 6% for CNN) is classified as hard soil.

For easier visualization, the results obtained from the CNN-based classifier are presented in Figure 13 during a sample straight run using semantic labelling where the successive terrain patches traversed by the rover are marked according to a color map that

TABLE 10 Category of difficulty assigned to each terrain type of the training set

Terrain type	Equivalent category of terrain difficulty
Sand	High
Gravel	Medium
Paved Ground	Low

TABLE 11 Terrain difficulty predictions as obtained from SVM and CNN in the sand mine test

Terrain difficulty labels	SVM	CNN
High	215	211
Medium	52	73
Low	35	18

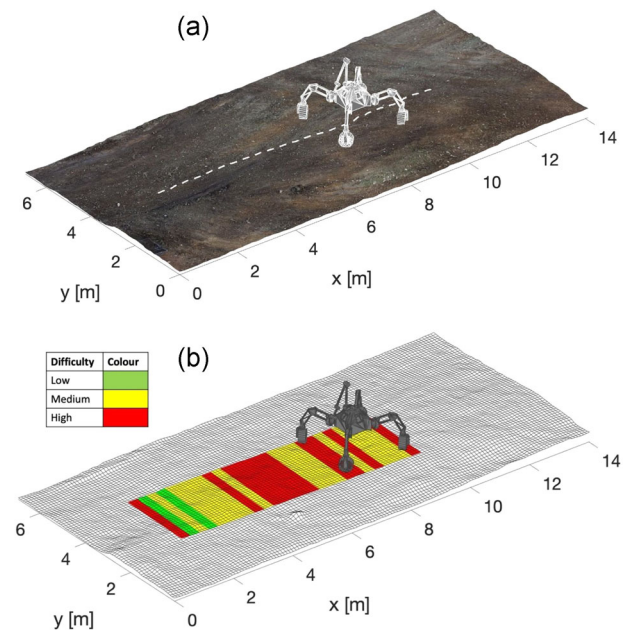


FIGURE 13 Semantic labeling using discrete terrain difficulty categories: (a) 3D stereo-generated map of the environment with overlaid the path (dashed white line) followed by the rover, (b) corresponding terrain difficulty visualization. Terrain patches are marked respectively in red, yellow, and green, for high, medium, and low difficulty [Color figure can be viewed at wileyonlinelibrary.com]

reflects the terrain difficulty scale of Table 10 (see also to the inset of Figure 13b). We recall that three discrete levels of terrain difficulty are considered: low, medium, and high.

Figure 13a shows the 3D stereo-generated map of the environment with overlaid a CAD model of SherpaTT and the path followed by the rover denoted with a dashed white line, whereas in Figure 13b the corresponding terrain labeling is reported with terrain patches marked respectively in red, yellow, and green, for high, medium, and low difficulty. In this test that was performed on fairly homogeneous terrain, the system mostly classifies the sand mine surface as of medium-high difficulty with two erroneous predictions (low difficulty) between 2 and 3 m.

6 | CONCLUSIONS

This study presented an approach to soil classification that relies on proprioceptive sensing only, for example, accelerations, forces, torques, and electrical currents. The algorithms developed are validated on data collected during tests performed with the hybrid wheeled-legged rover SherpaTT. The physics-based signal augmentation process presented in this paper uses 11 proprioceptive measurements to produce a large set of 80 features for SVM and 20 signals for CNN. This improved the information content as proved by the high classification accuracy obtained in generalization (90.8% for SVM and 96.4% for CNN). The proposed feature selection algorithm allows SVM to retain a high classification accuracy with only a portion of the full set (18 features), with successful reductions in memory usage (−77%) and required time for training (−6%), testing (−29%), and feature extraction (−33%). The same benefits also apply for CNN when using a reduced set of 13 signals related to the 18 best SVM features, improving memory usage (−14%), training time (−27%), testing time (−4%), and feature extraction time (−41%). The comparison between SVM and CNN shows good capabilities of both models in generalization, with accuracy higher than 90%. More challenging extrapolation problems have been tackled as well to evaluate the impact of varying operating conditions and site of the acquisition. In these tests, CNN outperformed the SVM counterpart. When tested on a new vehicle velocity, CNN reached an accuracy of 89.5%, against 55.7% held by SVM. When tested on new terrain, CNN recognized its deformability class more frequently than SVM, correctly classifying 6% more of the available samples. Based on these results, the proposed CNN qualifies as a good algorithm for soil classification even in the presence of disturbances and unknown conditions.

This study proved that is possible to use only proprioceptive features to infer the signature of a particular surface via learning algorithms. Moreover, the presented promising results suggest the possibility to extend rover traveling distance thanks to on-board integration of the developed learning algorithms.

Future developments of this study refer to (i) continuous training of the system by incorporating instances of “new terrain” classes during normal operations, therefore making the system adaptive, (ii)

augmenting the classifier with new special classes; for example, instances of excessive wheel slippage (close to 100%) can be used to train a hazard class to inform the rover of impending immobilization conditions, (iii) combining the proposed framework using proprioceptive signals with exteroceptive signals. The latter would enable the vehicle to predict hazards or trapping conditions before driving through the ground, for example, based on noncontact information coming from vision sensors.

ACKNOWLEDGEMENTS

The financial support of the projects: Autonomous Decision making in very long traverses (ADE), H2020 (Grant no. 821988), Agricultural in Teroperability and Analysis System (ATLAS), H2020 (Grant no. 857125), and multimodal sensing for individual pLANT phenOTyping in agriculture rObotics (ANTONIO), ICT-AGRI-FOOD COFUND (Grant no. 41946) is gratefully acknowledged.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

SOFTWARE REPOSITORY

The codes and data used for this study will be made publicly available at a Github repository.

ORCID

Angelo Ugenti  <http://orcid.org/0000-0002-8396-3966>
 Fabio Vulpi  <http://orcid.org/0000-0002-0216-8398>
 Raúl Domínguez  <https://orcid.org/0000-0002-0191-8345>
 Florian Cordes  <https://orcid.org/0000-0001-7966-8334>
 Annalisa Milella  <http://orcid.org/0000-0001-9456-7590>
 Giulio Reina  <http://orcid.org/0000-0003-1793-4419>

REFERENCES

- Bai, C., Guo, J., Guo, L., & Song, J. (2019). Deep multi-layer perception based terrain classification for planetary exploration rovers. *Sensors (Switzerland)*, 19(14), 3102. <https://doi.org/10.3390/s19143102>
- Barnes, D., Maddern, W. & Posner, I. (2017). Find your own way: Weakly-supervised segmentation of path proposals for urban autonomy. In *Proceedings—IEEE International Conference on Robotics and Automation* (pp. 203–210). <https://doi.org/10.1109/ICRA.2017.7989025>
- Bellone, M., Reina, G., Caltagirone, L., & Wahde, M. (2018). Learning traversability from point clouds in challenging scenarios. *IEEE Transactions on Intelligent Transportation Systems*, 19(1), 296–305. <https://doi.org/10.1109/TITS.2017.2769218>
- Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8), 1798–1828. <https://doi.org/10.1109/TPAMI.2013.50>
- Brooks, C. A., & Iagnemma, K. (2005). Vibration-based terrain classification for planetary exploration rovers. *IEEE Transactions on Robotics*, 21(6), 1185–1191. <https://doi.org/10.1109/TRO.2005.855994>
- Cordes, F., & Babu, A. (2016). SherpaTT: A Versatile Hybrid Wheeled-Leg Rover. In *Proceedings of the 13th International Symposium on Artificial Intelligence, Robotics and Automation In Space (ISAIRAS-16)*.

- Cordes, F., Kirchner, F., & Babu, A. (2018). Design and field testing of a rover with an actively articulated suspension system in a Mars analog terrain. *Journal of Field Robotics*, 35(7), 1149–1181. <https://doi.org/10.1002/rob.21808>
- Cowen, R. (2005). Opportunity rolls out of Purgatory. *Science News*, 167(26), 413.
- Dietterich, T. G., & Bakiri, G. (1994). Solving multiclass learning problems via error-correcting output codes. *Journal of Artificial Intelligence Research*, 2, 263–286. <https://doi.org/10.1613/jair.105>
- Dimastrogiovanni, M., Cordes, F., & Reina, G. (2020). Terrain estimation for planetary exploration robots. *Applied Sciences*, 10(17), 6044. <https://doi.org/10.3390/app10176044>
- Duan, K. B., & Keerthi, S. S. (2005). Which is the best multiclass SVM method? An empirical study. *International Workshop on Multiple Classifier Systems*, 3541, 278–285. https://doi.org/10.1007/11494683_28
- DuPont, E. M., Moore, C. A., Collins, E. G., & Coyle, E. (2008). Frequency response method for terrain classification in autonomous ground vehicles. *Autonomous Robots*, 24(4), 337–347. <https://doi.org/10.1007/s10514-007-9077-0>
- Dutta, A., & Dasgupta, P. (2017). Ensemble learning with weak classifiers for fast and reliable unknown terrain classification using mobile robots. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(11), 2933–2944. <https://doi.org/10.1109/TSMC.2016.2531700>
- ESA. (2021). Robotic exploration of Mars. <http://exploration.esa.int/mars/>
- Giguere, P., & Dudek, G. (2011). A simple tactile probe for surface identification by mobile robots. *IEEE Transactions on Robotics*, 27(4), 534–544. <https://doi.org/10.1109/TRO.2011.2119910>
- Gonzalez, R., Chandler, S., & Apostolopoulos, D. (2019). Characterization of machine learning algorithms for slippage estimation in planetary exploration rovers. *Journal of Terramechanics*, 82, 23–34. <https://doi.org/10.1016/j.jterra.2018.12.001>
- Guo, J., Guo, T., Zhong, M., Gao, H., Huang, B., Ding, L., Li, W., & Deng, Z. (2020). In-situ evaluation of terrain mechanical parameters and wheel-terrain interactions using wheel-terrain contact mechanics for wheeled planetary rovers. *Mechanism and Machine Theory*, 145, 103696. <https://doi.org/10.1016/j.mechmachtheory.2019.103696>
- Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3, 1157–1182. <https://doi.org/10.1162/15324430322753616>
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning: data mining, inference, and prediction. In *Springer Science & Business Media* (Second).
- Ishikawa, R., Hachiuma, R., & Saito, H. (2021). Self-supervised audio-visual feature learning for single-modal incremental terrain type clustering. *IEEE Access*, 9, 64346–64357. <https://doi.org/10.1109/ACCESS.2021.3075582>
- JPL. (2021). Mars exploration rovers. <http://marsrovers.jpl.nasa.gov/home/index.html>
- Kingma, D. P., & Ba, J. L. (2015). Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations (ICLR)*.
- Lin, Y., Lee, Y., & Wahba, G. (2002). Support vector machines for classification in nonstandard situations. *Machine Learning*, 46, 191–202. <https://doi.org/10.1023/A:1012406528296>
- Lu, D., & Weng, Q. (2007). A survey of image classification methods and techniques for improving classification performance. *International Journal of Remote Sensing*, 28(5), 823–870. <https://doi.org/10.1080/01431160600746456>
- Manduchi, R., Castano, A., Talukder, A., & Matthies, L. (2005). Obstacle detection and terrain classification for autonomous off-road navigation. *Autonomous Robots*, 18(1), 81–102. <https://doi.org/10.1023/B:AURO.0000047286.62481.1d>
- Nampoothiri, M. G. H., Vinayakumar, B., Sunny, Y., & Antony, R. (2021). Recent developments in terrain identification, classification, parameter estimation for the navigation of autonomous robots. *SN Applied Sciences*, 3, 480. <https://doi.org/10.1007/s42452-021-04453-3>
- Nasa, Mars 2020. (2020). <https://mars.nasa.gov/mars2020/>
- Ocón, J., Dragomir, I., Coles, A., Green, A., Kunze, L., Marc, R., Perez, C. J., Germa, T., Bissonnette, V., Scalise, G., Foughali, M., Kapellos, K., Domínguez, R., Cordes, F., Paar, G., Reina, G., & Kisdi, A. (2020). Ade: Autonomous decision making in very long traverses. In 15th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS'20).
- Otsu, K., Ono, M., Fuchs, T. J., Baldwin, I., & Kubota, T. (2016). Autonomous terrain classification with co-and self-training approach. *IEEE Robotics and Automation Letters*, 1(2), 814–819. <https://doi.org/10.1109/LRA.2016.2525040>
- Reina, G., Leanza, A., & Messina, A. (2020). Terrain estimation via vehicle vibration measurement and cubature Kalman filtering. *Journal of Vibration and Control*, 26(11–12), 885–898. <https://doi.org/10.1177/1077546319890011>
- Reina, G., Milella, A., & Galati, R. (2017). Terrain assessment for precision agriculture using vehicle dynamic modelling. *Biosystems Engineering*, 162, 124–139. <https://doi.org/10.1016/j.biosystemseng.2017.06.025>
- Tai, L., Li, S., & Liu, M. (2017). Autonomous exploration of mobile robots through deep neural networks. *International Journal of Advanced Robotic Systems*, 14(4), 172988141770357. <https://doi.org/10.1177/1729881417703571>
- Vapnik, V. N. (2013). The nature of statistical learning theory. In *Springer Science & Business Media* (Second). <https://doi.org/10.1007/978-1-4757-2440-0>
- Vulpi, F., Milella, A., Marani, R., & Reina, G. (2021). Recurrent and convolutional neural networks for deep terrain classification by autonomous robots. *Journal of Terramechanics*, 96, 119–131. <https://doi.org/10.1016/j.jterra.2020.12.002>
- Wellhausen, L., Dosovitskiy, A., Ranftl, R., Walas, K., Cadena, C., & Hutter, M. (2019). Where should i walk (Predicting terrain properties from images via self-supervised learning. *IEEE Robotics and Automation Letters*, 4(2), 1509–1516. <https://doi.org/10.1109/LRA.2019.2895390>
- Zhao, Q., & Fränti, P. (2014). WB-index: A sum-of-squares based index for cluster validity. *Data & Knowledge Engineering*, 92, 77–89. <https://doi.org/10.1016/j.datak.2014.07.008>

How to cite this article: Ugenti, A., Vulpi, F., Domínguez, R., Cordes, F., Milella, A., & Reina, G. (2021). On the role of feature and signal selection for terrain learning in planetary exploration robots. *Journal of Field Robotics*, 1–16. <https://doi.org/10.1002/rob.22054>