MULTI-MODAL SENSOR FUSION IN TERRAIN CLASSIFICATION

By

MARCOS JAY T. CONCON

CURTIN UNIVERSITY MALAYSIA
School of Electrical and Computer Engineering

JUNE 2020

June 5, 2020

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter One

Introduction

1.1 Background

Aristotle may have been the first to describe how automated mechanical statues could replace slaves and reduce the burden of everyday labor. From automating automotive manufacturing processes in the 1970's, robotics coupled with rapid advancements in computer science, physics, mathematics and mechanical engineering have now democratized the deployment of these automated systems in our daily lives. For example, Amazon is a compnay that employs over 200,000 mobile robots in its warehouse network, working side by side with thousands of employees to accomplish fast and efficient deliveries to their customers. Robotic applications are not solely focused on highly-controlled factory applications but in more unconstrained and complex environments. Meticulous input regarding dynamic environments is thus needed to fully understand and evaluate a robots position and path navigation towards an objective location [1].

1.2 Motivation and Design

There is a challenge for safe autonomous robot navigation as they are increasingly deployed in unstructured and semi-structured environments such as forestry, mining, rescue, space exploration and site inspection. Due to the high degree of uncertainty encountered in each of these different environments, the robot must be capable in the perception and interpretation of the environment in a meaningful way [2]. Limitations in current sensing technology, difficulties in modelling the interaction between robot and terrain, and a dynamically changing unknown environment all makes this difficult [3]. Mobile robots traversing safely at high speed in hard flat ground may experience slippage, sinking, and embedding events in the face of loose slippery terrains [4], [5]. Other surfaces can be bumpy and rocky, which may result in damage to the robot [6]. Consequently, the terrain itself can become a hazard, referred to as a nongeometric hazard [7]. To achieve efficient and safe navigation, a mobile robot should adapt its driving style, control strategy, or path planning strategy to accommodate characteristics of the terrain and this relies on its capability to recognize or classify its surrounding environment.

Terrain classification is generally categorized as vision-based (exteroceptive), reaction-based (proprioceptive) or a combination of both. Vision and reaction-based approaches are analogous to a human driver's recognition of a terrain based on what is seen visually and felt through the robots reactions during traversal of the terrain. Vision-based terrain classification is typically performed using cameras or laser range finders. Some visual techniques simply provide detail about the surrounding environment and not the classification of the traversed terrain. These include three dimensional maps to show navigability [8], vegetation, shrubs and trees [9, 10], detection of unexplored terrains using stereo imagery [11], and detection of surface features from image processing. Research has also shown visual detection from aerial vehicles as an effective means of recognizing topology, which can be used for

planning and detection of road conditions [12].

Reaction-based classification relies on proprioceptive sensor measurements such as wheel slip, wheel sinkage and robot wheel vibrations while the mobile robot is in operation and is also known as terrain classification using proprioceptive sensors. However, as previously noted, some robot terrain interactions such as slip and wheel sinkage can be difficult to measure accurately. Thus reaction based terrain classification is most often performed using vehicle vibrations, as it is easy to measure using inertial sensors and accelerometers. Additionally, other measures can be used to possibly enhance the effectiveness of vibration based terrain classification. The theory and physical basis behind the vibration-based terrain classification pattern recognition structure used for automated reaction-based terrain classification is given in Chapter 2. Terrain classification that is robust to a wide variety of terrains, environmental conditions, and mobile robot operating conditions will almost certainly require the symbiosis of vision-based and reaction-based methods. This requires effectively integrating the two into a single classification system. There are a few works that have been sought to address this issue of classifier fusion which is described in Chapter 2. Subsequent work in the area of classifier fusion should prioritize analyzing how these fusion techniques perform in circumstances that would normally cause either vision- or reaction-based classification to fail.

1.3 Research Significance

Research conducted on individual approaches towards terrain classification using either vision or reaction-based systems have been plentiful which are all discussed in Chapter 2. There are however, very few studies on classification based on data fusion where complementary information from multiple sensors can be taken advantage of to obtain a more robust overall

accuracy. Specific areas where current research lacks are classification models generated from the merging of vibration, acoustics and image data. Moreover, an evaluation of the computational performance and accuracy of deep learning methods for multiple sensor feature extraction versus manually extracted features have yet to be accomplished. Therefore, this study wishes to explore and evaluate the results and performance of multi-sensor fusion in classifying terrain types with a deep learning approach.

1.4 Aims and Objective

Aim

The aim of the research is to investigate, apply and benchmark multiple sensor fusion techniques in classifying different terrain types. Moreover, a feature learning approach on multiple sensor data to classify terrain types is to be evaluated in terms of computational performance against traditional feature extraction methods.

Objectives

1) Develop classifier models to evaluate the effectiveness of combining several sensor features such as acoustics, vibration and textures towards terrain classification.

2) Study the effects of feature learning approaches and evaluate its accuracy and computational performance against current approaches utilising handcrafted features.

3) Deploy a suitable model on a realtime mobile robot to validate the conclusions made towards terrain classification in this research.

Chapter Two

Literature Review

This section evaluates existing current trends in the field of terrain classification for autonomous mobile-robot navigation. Section 2.1 presents an overview of the navigational and control requirements of an autonomous mobile robot. Sections 2.2, 2.3 and 2.4 lists and discusses different approaches towards terrain classification with Section 2.5 summarising the research gap.

2.1 Robot Navigation

Robotic navigation is the process of evaluating one's position, planning and following a path in its current environment. A universal requirement for almost all mobile robots is that they should possess the ability to find its way in an environment. In certain scenarios such as surveillance or cleaning operations, the ability to purposefully navigate in a structured or unstructured environment may not be required but is a necessity in the most scientific or industrial-based applications. Thus, a robot's capability to autonomously navigate plays a distinct role in its success and is the baseline for all relative technologies of autonomous mobile robots.

There are three categories where robotic navigation can be split into to better comprehend the problems faced in each instance. They are described as follows 

2.1.1 Localization - Where am I?

Localization is the process of estimating where the robot is, relatively to some model of the environment, using whatever sensor measurements are available. A continuously moving robot has a tendency of introducing drift or change in the estimation of its location in which it has to be kept updated through a series of active computation [14]. The updates are determined from the identification of special landmark features, characteristic signals from sensor measurements and probabilistic models. The problem of localization requires information from relative and absolute measurements which return feedback about its driving actions and the state of its surrounding [15]. However, uncertainty arises from measurement noise from real-world sensors leading to inaccurate localizations. Several probabilistic methods have been employed to tackle localization problems stemming from uncertainty characteristics [16] . The two most common are Markov Localisation [17] and Particle Filters [18].

2.1.2 Mapping - Where am I going?

The mapping issue exists when the robot does not have a guide of its condition and steadily constructs one as it navigates its environment. During navigation, the robot distinguishes key landmark features  ich are registered for use regarding its environment. The main concern for the mapping problem is the mobile robots perception of the environment.

To locate obstacles, detect distances and observe landmarks, the mobile robot must have

a sensing mechanism that enables measurement collection. The sensors used for mapping depend on the type of mapping that needs to be done. Most common sensors are sonar, digital cameras and range lasers. Approaches for mapping have been accomplished considering the extraction of natural features from the environment [18] and through the identification of special artificial landmarks [19]. The complexity of the mapping problem is the result of a different number of factors [16], the most important of which are: Size of the environment; noise in perception and actuation; perceptual ambiguity and cycles.

2.1.3 Path planning - How do I get there?

Path Planning is the process of looking ahead at the outcomes of the possible actions, and searching for the sequence of actions that will drive the robot to the desired goal [14]. It involves finding a path from the robot's current location to the destination. The cost of planning is proportional to the size and complexity of the environment. The bigger the distance and the number of obstacles, the higher the cost to the overall planning. The cost of planning is a very important issue for real-time navigation needs, as the longer it takes to plan, the longer it takes to find a solution. Path Planning techniques for navigation can be divided into two subcategories:

- Local Path Planning are solutions that do not imply much complexity since they use only local information of the environment. They often do not offer optimal solutions and also have the common problem of local minima.
- Global Path Planning takes into account all the information of the environment at the same time. Unfortunately this type of planning is not appropriate for real-time obstacle avoidance because of the high processing needs for all the environment's data.

There are many different approaches to path planning which try to solve the problem

using different techniques. Two relevant Path Planning techniques are the Artificial Potential Field [20] and approaches based on the Ant Colony [21].

2.1.4 SLAM

SLAM (Simultaneous Localization and Mapping) is a technique that involves both localization and mapping to construct a map for the autonomous mobile robot to use within an unknown environment. This map also allows it to keep track of their current position. This approach is complex since it involves both localization and mapping processes both with uncertainties associated. One main concern in SLAM is keeping the uncertainty, for both robot position and landmark position, controlled, thus keeping errors as low as possible. For this double uncertainty, SLAM usually uses Kalman Filter and Particle Filter methods. SLAM is briefly discussed here but is not explained in detail in this research as it achieves a different purpose in terrain applications by focusing on segmenting ground surfaces from different obstacles rather than classifying the traversed terrain.

2.2 Approaches involving exteroceptive modalities

Methods employing exteroceptive modalities for terrain classification are usually realized by optical sensors (e.g digital camera, infrared camera and LiDAR), which have been widely investigated. Non-interactive methods possess high classification accuracy, but they falter in conditions such as illumination, variations in appearance or cover such as leaves.

2.2.1 Local Binary Patterns (LBP)

Local Binary Patterns (LBP) are a simple, computationally efficient yet powerful approach towards describing terrain texture. A basic LBP utilises a 3x3 window on each pixel of a grey-scale image in which all neighbouring pixels are thresholded by a value k based on its value and the center pixel c. A value of 0 is assigned to neighbouring pixels that possess a lower weighting than the center and a value of 1 assigned otherwise. The normalised values of 1's or 0's are then joined together to form a binary pattern that is used generate a LBP code which describes the texture at that specific pixel. Since the 8-bit binary pattern can have 256 values, a histogram containing 256 dimensions for classification is generated [22]. The LBP encoding process is shown in Figure 2.1.

Applications in research and their results

A comparative study done by Yasir, evaluated the performance and accuracy of a derived LBP descriptor called the Local Ternary Patterns (LTP) which was used to classify 5 outdoor terrain textures from a camera mounted on a quadcopter. Instead of generating a binary pattern, the LTP generates a ternary pattern using a threshold k around the value c of the center pixel. Pixels around the center which possess a value greater than $c+k$ are given a value of 1, those with value less than c-k are assigned -1 and values in between them are

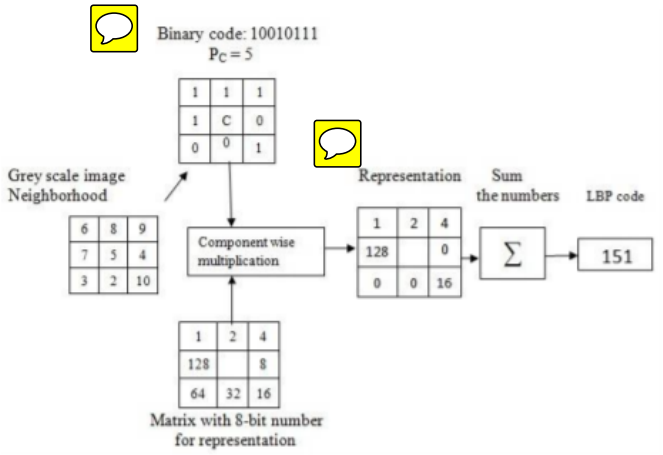

Figure 2.1 Local Binary Patterns Encoding Process

assigned 0 [23].

$$D_{it} = \begin{cases} 1 & T \geq (c+k) \\ 0 & T < (c+k) and T > (c-k) \\ -1 & T \leq (c-k) \end{cases} \tag{2.1}$$

Two distinct LBP codes are generated from the separation of the obtained ternary pattern where one segment contains all the positive values and the other containing all the negative ones. Two separate histograms are computed from the two LBP code and are then concatenated together to form a histogram of 512 dimensions.

Among all other classifiers, the results obtained from using the Random Forest classifier were only described in the research as it produced the best overall result. The Random Forest is primarily used in classification problems by building multiple decision trees based on image data and merging them together to obtain a more accurate and stable prediction. For a 640 x 480 image divided into 100 x 100 patches, the mean classification accuracies for

6 terrain types were 79.5% and 84.4% for Local Binary Patterns and Local Ternary Patterns respectively. However, the performance metrics for timing the cross-validation and classification were better for LBP, taking about 185 seconds compared to 260 seconds for LTP.

A real-time approach was done by Nantheera to influence a robot's locomotion from the terrain classification results of three categories which were hard surfaces (eg. tarmac, brick, tiles), soft surfaces (eg. grass, soil, gravel) and unwalkable areas (eg. static and moving obstructions) [24]. Notably, uncertainties such as high movement or consecutive motion blur of the captured video frames were considered during the experiment.

The pipeline that resulted in the best texture classification performance first included the *intensity level distribution*, where the mean, variance, skewness, kurtosis and entropy were extracted. The *wavelet transform* employing the Dual-Tree Complex Wavelet Transform (DT-CWT) was used to extract 48 features from both spatial and frequency information. The DT-CWT is a two-dimensional wavelet transform which provides multiresolution, sparse representation, and useful characterization of the structure of an image. To improve the overall performance for the pipeline, the uniform pattern was obtained from the Local Binary Patterns to generate a reduced histogram dimensionality of 59 bins where each bin was used as 1 feature. As stated in Nantheera's research, a high proportion of the texture patterns can be found from this uniform pattern.

The mean classification accuracies for 15 test videos containing all types of terrains over a walk of duration 40 was 82%, an improvement of 15.3% from [25] (an approach that solely relied on colour and textural features). However, comparing [25], the overall pipeline increases computational time by 8%.

2.2.2 Bag of Visual Words (BOVW)

The Bag of Visual Words has emerged as a robust paradigm for visual terrain classification. The framework produces a semantic representation with texture descriptors for accurate visual terrain classification. The first core step in the BOVW framework relies on Feature Extraction where low-level local features are detected and obtained from terrain images to define keypoints and descriptors of an image such as color and texture. Many local feature detection algorithms such as ColorHist, GIST, SIFT, SURF, FCTH and JCD have been employed in visual terrain classification [26].

The second core step is Codebook Generation. Codewords which are characteristic representations of the image descriptor are created by encoding local features obtained from the initial Feature Extraction phase. A collection of codewords is referred to as a codebook And are considered to be characteristically representative of the image descriptors [27]. The following two techniques are often used during the codebook generation stage:

- K-means-clustering: partitioning the local descriptor space into informative regions (codewords), each of which is represented by its center.
- Gaussian Mixture Model Clustering: using the generative models to capture the probability distribution of the local descriptors.

Feature Coding, the next core step in the BOVW framework uses the codebook to map the descriptor space into a coding space. Typically, different coding methods would construct different coding spaces and thus yielding different discrimination representations. Coding methods can be divided into 2 types: Activation Based Encoding and Difference Based Encoding. The essential difference between these 2 coding approaches is the way the information is obtained from the descriptor space as shown in Figure 2.2.

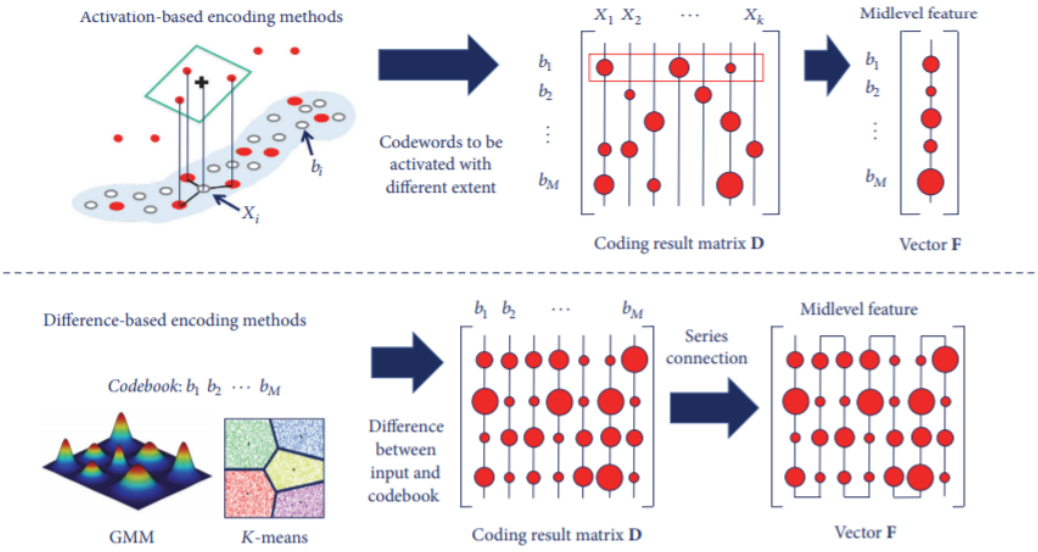

Figure 2.2 Feature coding approaches in the BOVW framework [28]

Pooling and Normalization methods are then applied to the Feature Coding result to produce a compact global image-level representation which is then fed to a classifier such as Support Vector Machine (SVM) for terrain classification. The entire BOVW framework can be summarised in Figure 2.3.

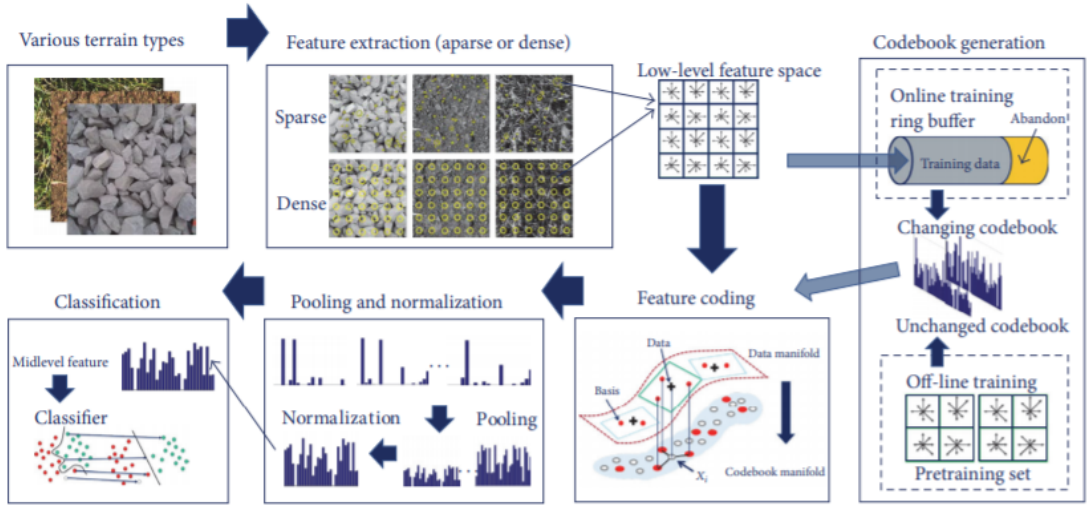

Figure 2.3 The BOVW Framework [28]

Applications in research and their results

In Pauls work [29], real-time classification for four terrain types was performed on a small-legged robot mounted with a single-compact camera. A different gait pattern was initiated when the legged robot classified certain terrain types in its test environment. Their methodology involved the Bag of Visual Words created from speeded up robust features (SURF) with a support vector machine (SVM) classifier.

Feature extraction was performed by first computing texture descriptors from detecting key points at unique locations on the image with the SURF algorithm. A visual vocabulary for the BOVW framework was generated with the use of K-means clustering which was used to locate the number of centers that best describe groupings of descriptor data. Each image was then described by a word frequency vector from the generated visual vocabulary for that image. Conducted tests in an experimental terrain showed that a 100% verification accuracy was obtained from gait patterns initiated by the classified terrain type. In the best performance-case scenario, the legged robot took 20 seconds to identify grass textures whilst in the worse case, took up to 45 seconds to identify chips/big rocks. It is important to note however that the classification results in their work do not have metrics for confidence measurements.

Recently, an optimised pipeline for terrain classification was proposed in [28] that involved a Principal component analysis (PCA) whitening technique, an improved Bag Of Visual Words framework and the average kernel fusion method. PCA analysis optimises raw input descriptors into much lower dimensional descriptors resulting in a dimensional reduction while incurring notably little error. A whitening technique was implemented in tandem with the PCA analysis to ensure that the input is less correlated and that it possesses the same variance. Decorrelating the descriptor, reducing the dimensions and normalizing the

variance in the pre-processing stage contributed to a performance improvement of terrain classification in their experiments. The GIST descriptor based on Gabor filters was used to generate global features which are then combined with the midlevel features from power L2 normalization. The average kernel fusion method was then used to combine the global and midlevel features to enhance performance and to complete the terrain classification task effectively.

The improved Bag of Visual Words framework used in tandem with the feature fusion method achieved a mean classification score of 89% when applied to the DS1 dataset containing over 8 different texture classes. The optimised pipeline took 454 seconds to classify 1200 terrain images in the actual run time which involved feature extraction, encoding, pooling, normalization and fusion.

2.2.3 Convolutional Neural Networks (CNN)

Deep learning is an emerging trend in every area of engineering and science. It has been used in various fields from diagnosis of certain diseases in the medical field, to astronomy where exoplanets are being discovered [30]. In engineering, it has been used as a pioneering choice in autonomous self-driving vehicles, in speech and face recognition [31]. Convolutional Neural Networks (CNN) are one of the most successful architectures in the field of Deep Learning and is an increasingly popular mechanism for image analysis in near and far-range terrain classification.

A CNN is a particular type of neural network that manipulates the input image (or input signal) to obtain certain features. Convolutional layers in the neural network act as a bank of filters and so several layers can be stacked as frequently as filters are desired to apply in order to highlight more sophisticated characteristics of the input [32]. A pooling layer is then placed after the stack of convolutional layers of which its purpose is to discard all other neurons except those with the maximum activation value in each grid. Activation values are simply numerically expressed information. The next layer is the dropout layer where regularization is applied to reduce overfitting. The dropout layer does this by dropping some neurons and their corresponding inputs and outputs connections randomly and periodically to ensure that each neuron "learns" something useful for the network [33]. Finally, a set of fully-connected (dense) neural networks are added to the architecture [32] with the final layer returning the class label for each input image. Different architectures can be produced by cascading the above-mentioned layers in various ways.

Applications in research and their results

The CNN shown below was used in Gonzalez's research in validating its performance against traditional machine learning methods such as the Support Vector Machinein terrain classification. In the case for traditional machine learning methods, a global descriptor of the images and a filter of the signals from the proprioceptive sensor is required to achieve satisfactory classification performance. However, CNN's can take a different approach in that the network itself can extract meaningful features directly from fed raw data while still achieving good classification accuracy [33]. This is an advantage as time can be saved in trying to design a pre-processing filter to extract important features of the signal or image.

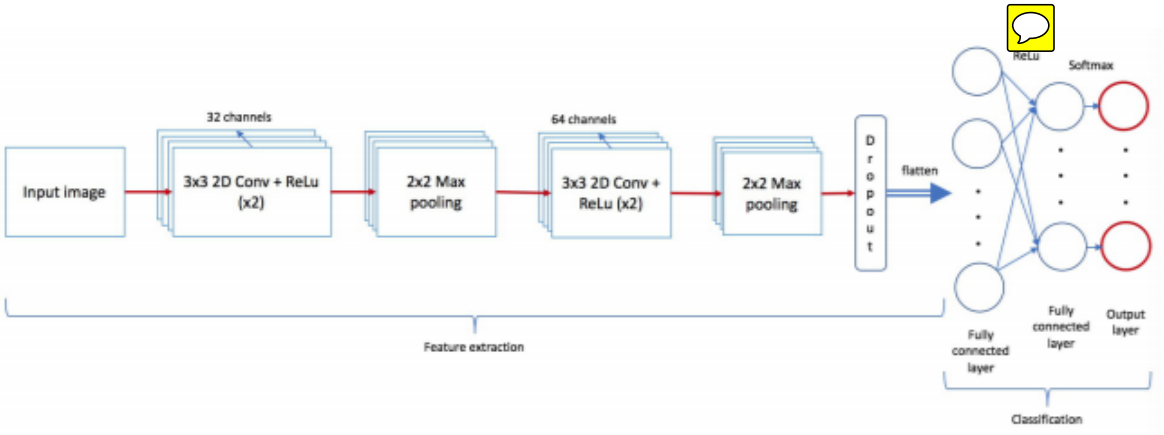

Figure 2.4 A typical CNN pipeline [34]

From an average of ten independent runs where various sizes for the training/testing sets have been considered (e.g. 70/30%, 60/40%, 50/50%), the CNN algorithm trained on raw images achieved an 87% mean accuracy in classifying 11 texture types versus 92% for the Support Vector Machine supplemented with the HOG image filter. Without the filter, however, the Support Vector Machine scored an abysmal 10% accuracy. One other important remark from Gonzalez's experiment is that once the CNN is trained, the testing time is similar to other machine learning algorithms (of the order of seconds).

More recently, the implementation of a 3D CNN was evaluated by Zhang against the performance of a 2D CNN classifying far-range terrain types from aerial Polarmetric Radar Imaging data. Compared with 2D CNN's as discussed previously, the input of the 3D CNN is a cube stacked with multiple feature maps, which extracts features at three scales at the same time. Through 3D convolution kernel, the features can be extracted from continuous multiple feature maps, and the feature cube can be connected to multiple consecutive feature maps in the upper layer [36]. This process is shown in Figure 2.5 below.

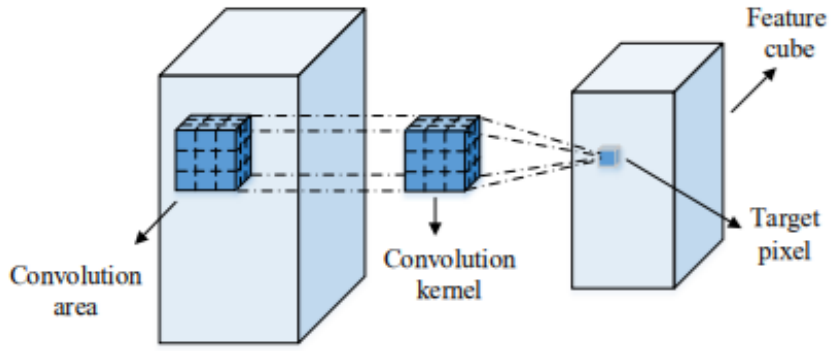

Figure 2.5 3D Convolution Process [35]

In the experiment, both architectures were tuned with a 2 layered convolutional layer, 2 max pooling layers, a fully connected layer after the second max pooling layer and the softmax layer as the final output. The Rectified Linear Unit (ReLU) was used in both architectures which is an activation function for the outputs of the CNN neurons. For polarimetric SAR data of an area containing 15 identifiable classes, the average classification accuracy obtained for the 3D CNN was 95.29% versus 93.55% for 2D CNN. There was no mention however, of the computational cost and time from cross-validation to testing for 3D CNN's. In a different research field [37], the evaluation of the computational performance & classification accuracies of different CNN architectures was done on MRI images, the 3D CNN achieved a higher mean score but was found to be more computationally costly due to the extra dimensions added to the input.

2.3 Approaches involving proprioceptive modalities

Methods employing proprioceptive modalities have not been investigated in the same depth as visual-based methods but there is a sizeable amount of work in this area. Current approaches usually realized by acoustics, haptics and vibration are less computationally costly while being more robust to changes in environmental lighting.

2.3.1 Acoustic based

Acoustic terrain classification relies on sound interactions between the robot and terrain during locomotion pressing on ground surface. Acoustic terrain classification first stemmed from [38], where methods from research domains such as speech recognition were used to extract temporal and spectral characteristics with their efficacy verified within terrain classification. A typical acoustic classification pipeline involves collection of audio data of robot locomotion during traversal in various terrain settings. Pre-processing which involves windowing, audio inspection and noise-removal are applied on the acquired data.

In acoustic based applications, feature extraction involves a transformation of the acquired audio signal from a microphone into the frequency domain from the time domain. This is usually realized by the Discrete Fourier Transform or the more computationally efficient Fast Fourier Transform. Along with time-domain characteristics, features obtained from the signal power spectrum such as spectral flux, centroid, roll-off, spread, kurtosis and short-time energy (STE) could be used in classification [39]. The Mel-frequency Cepstral Coefficients which collectively make up a representation of the short-term power spectrum of the sound are also a reliable feature used in acoustic-based processing. These features are then fed to a classifier, typically a Support Vector Machine to generate a classification model.

Applications in research and their results

Acoustic vehicle-terrain interaction has been the least explored among all the other proprioceptive methods. In Libby's research, utilizing spectral coefficients, moments and various other spectral and temporal features for sound-based classification, a mean classification score of 78% was achieved for six terrain types with a Support Vector Machine (SVM) classifier improving to 92% when smoothing is applied over a two second temporal window [38]. In recent research conducted by Christie, an SVM classifier trained on a dataset containing 5 minutes worth of data for seven terrain types utilised statistical spectral and band characteristics in a 1 second window to achieve a mean of 95.1% after servo noise removal by spectral subtraction [40]. In both of these approaches however, the classifier was trained on comparatively limited data with manual feature extraction after selective pre-processing. There are two key issues identified with these approaches:

- manual feature extraction for every scenario is impractical
- they often lack reliability in real-world scenarios

In [39] , a robust proprioceptive terrain classification system was introduced that sought to address these issues. The pipeline consists of an optimised Recurrent Convolutional Neural Network with a noise awareness training scheme that was used to learn complex dynamics in unstructured audio data replacing the need for manual feature extraction. The model achieves a 97.36% mean classification score on the offline experiment for identifying 9 (indoor and outdoor) terrain types. This is a promising indication of the adoption of acoustic terrain classification in real world scenarios but various outdoor noises such as vehicles passing by are not accounted for in the experiment and this may lead to some inaccuracies.

2.3.2 Haptic based

Applications on haptics in describing different terrain types leverage on ground reaction forces by means of tactile sensors placed on the robot-terrain contact area. Thus, they are more applicable to legged robots than wheeled ones [41].

Multiple transduction techniques have been utilized to gauge contact forces and pressure distributions including capacitive, resistive, magnetic and optical means. Magnetic and optical based sensors which measure deflections in a thick, compliant skin are not well suited for robotic applications as they contribute too much bulk and weight to the system. Approaches dependent on optical fibres are challenging to adapt to robotics legs that experience continuous rotation. Moreover, piezoresistive sensors that rely on conductive inks or polymers may be economical and robust, but experience significant hysteresis, which limits their use for dynamic tactile sensing [42]. Pattern strain gages which are sensors strapped directly onto the feet are another possibility , but requires quality instrumentation amplifiers and there are required wiring constraints due to the continuously rotating legs and the robot body. In the present day, capacitive tactile sensors based on flexible circuits with surface-mounted CDCs are an emerging choice for haptic feedback due to their cost, low weight, robustness, and ability to wrap around various geometries [43]. For robots with rotary legs, the minimization of wiring is an additional advantage.

There is not a specific approach towards feature selection in classifying terrains with haptics due to the multitude of employable sensors which generate different characteristical traits. Typically for capacitive tactile sensors, features such as robot stride frequency, sensor force peak and average amplitudes, the robots motor torque peak and average amplitude are used in the classification process.

Applications in research and results

Hoffman's work investigated the effect of dynamic locomotion patterns with individual and combinations of tactile and joint angle sensors on a legged robot to evaluate the properties of multiple terrains. The activation of different gaits lead to different overall performance — some turn out to be more suited for perception [44]. In [45], for example, employ a similar approach by clustering data resulting from different actions of a robotic hand separately. Every coordinated motor pattern results in specific interaction with the environment and, consequently, in a specific spatio-temporal transformation of the incoming sensory data, which can be advantageous for subsequent perceptual tasks [46]. In Hoffman's work, the effect of coordinated vs. uncoordinated behavior on terrain perception was observed by the drop in performance when random motor commands were applied. Moreover, the combination of tactile and joint angle sensors was concluded to be a powerful sensory suite for terrain classification.

In recent developments, Alice's design of a miniature array of capacitive tactile sensors mounted on a flexible printed circuit was implemented with local signal processing and communications to measure the ground reaction forces for small legged robots. Using signal features from the sensors,the Support Vector Machine classifier was trained to perform single stride terrain classification with better than 90% accuracy on four representative types of terrain [47]. For the apparatus tested, the peak and average motor torque were directly related to the ground reaction forces and yielded the highest classifier feature scores. When this information is not available, or is not as well correlated with ground forces, using only the tactile sensor data and information about stride frequency was proven for reliable terrain identification in most cases.

2.3.3 Vibration based

Vibration signals are easily collected in contrast to haptic signals, since the classifier can perform well even with a consumer-grade accelerometer. Vibration-based terrain classification was first suggested in [47]. The idea is to measure the vibration that is induced in the robot while it traverses the terrain. The vibration can be measured at the wheels, the axes or the body of the robot. Usually, accelerometers are used to measure the vibration perpendicular to the ground surface. As different terrain types induce different vibration signals, one tries to learn characteristic vibration signals for each terrain type from training examples. The learned model is then used for classification of unknown data. The disadvantage of the method is that terrain can be classified only while the robot traverses it, but not beforehand. Advantages include independence from illumination conditions and the high reliability. Thus, vibration-based terrain classification can be used as a stand-alone classifier or in combination with other sensors [48].

Applications in research and their results

Early work done in [49] involves vibration for terrain classification and uses power spectrum as the feature with Principal Component Analysis (PCA) to reduce the dimensionality of their data and Linear Discriminant Analysis (LDA) for classification. In later work, researchers begin to develop more simple and compact features in the time domain instead of a more than 100-dimension frequency-domain feature [48]. An 8-dimensional feature vector composed of zero crossing rate, mean, standard deviation among other features is then developed costing less computing time than spectral methods; however, in some applications, these compact features are not sufficient to cover all the characteristics of the raw data. Support Vector Machine (SVM) has been demonstrated to be the best performing classifier using time- and frequency-domain features (weiss). In [50], Dynamic Cortex Memory

(DCM) which is an extension of Long Short Term Memory (LTSM) is addressed for vibration based terrain classification without any explicit feature computation. Their experiments on 14 terrain types achieves an overall accuracy of approximately 85%, which is the state-of-the-art accuracy for 14-class terrain classification. Moreover, a just-in-time computation implementation for their networks was used. Apart from terrain classification, vibration is also frequently used in damage detection field [51]. In this field, time-domain features with a specific goal, such as kurtosis, crest factor and Root Mean Square (RMS), along with frequency-domain features from the Discrete Fourier Transform (DFT).

2.4 Multi-modality approaches

Multi-modality methods combine several modalities based on their complementary characteristics, thus possessing a higher accuracy and robustness against environmental interference [52]. Early work on a multi-modality approach towards terrain classification was accomplished with Halatci's work for Mars Planetary Exploration Rovers. The results from classification algorithms for color, texture and range features were merged with the vibration features through Bayesian and the meta-classifier fusion to achieve 84% average accuracy for three terrain types as compared to 65% and 77% for pure vibration and pure color-based classification respectively.In [53], three-axis acceleration, roll-pitch-yaw (RPY), and angular rates combination with the bagging algorithm to achieve up to 63% better accuracy compared to an SVM classifier with less computation. Another approach in [54], five data sources including four vibration sources and one acoustic source are collected by their tracked robot for fusing predictions. A two-stage feature selection method that combines Relief and mRMR algorithms is developed to obtain optimal feature subsets. In addition, four different classifiers are combined for the classification task to achieve an average classification accuracy increase of 10% when benchmarked with SVM and KNN classifiers.

2.5 Summary of Research Gap

The literature review chapter reviewed the methodologies used for terrain classification. The design approaches for exteroceptive, proprioceptive and the fusion of both methods were evaluated along with emphasis on the computational costs and performance accuracies towards classfying different terrains. It is evident that the analysis and evaluation on sensor fusion methodologies with a feature learning approach is limited and hence it is necessary for more research to be done in this field for terrain classification. Table 1 below summarises the research conducted before and their outcomes:

Year, Author	Research Highlights	Remarks
2006, Weiss, Christian Fröhlich, Holger Zell, Andreas	Early vibration based terrain classification involving features from Fast Fourier Transform (FFT) and Power Spectral Density with a SVM classifier to classify 7 terrain types	Average accuracy of 91.7%
2011, Yasir	Evaluates the Local Binary Patterns (LBP) agaist it's derivative, the Local Terniary Patterns (LTP) with a Random forest classifier for 6 terrain types (grass, asphalt, gravel, big tiles, small-tiles, bushes)	Average accuracy of 79.5% (LBP) and 84.4 % (LTP)

Table 2.1 Research summary part 1

Year, Author	Research Highlights	Results and remarks
2012, PaulFil-itchkin andKatie Byl	Studies real-time robot gait activation fromterrain classification using a bag of visualwords (BOVW) created from speeded up robust features (SURF) with a support vectormachine (SVM) classifier. No metric for clas-sification accuracy but from gait activationonly.	100% verification accuracy-for gait activation from terrain type
2013, Ozkul	Adopts approaches for identfying 6 terrain types by deriving speech recognition literature with a feature set composing of spectral band energies augmented by their vector time derivatives and time-domain averaged zero crossing rate	78% average overall accuracy
2015, Nantheera	Implements an improved pipeline using uniform code from Local Binary Patterns (LBP) for idnetifying 3 terrain categories (hard surfaces, soft surfaces, unwalkable areas) containing multiple classes	Average accuracy of 82%
2016, Christie Kottege, Navinda	Implements an online real-time legged robot to classify 5 terrain types (carpet, concrete, grass, mulch, gravel) that utilise a 32 dimensional feature vector from the acoustic signals produced during locomotion.	92.9% sensitivity (true-positive rate)

Table 2.2 Research summary part 2

Year, Author	Research Highlights	Remarks
2016, Otte, Sebastian Weiss, Christian Scherer, Tobias Zell, Andreas	Adopts a vibration-based Recurrent Neural Network for classifying 14 terrain types without explicit feature computation	85% overall accuracy
2016, Wu, X. Alice Huh, Tae Myung Mukherjee	Evaluates tactile sensing data, in combination with information about the motor torque and robot gait, to distinguish among hard, slippery, grassy and granular terrain types	with $>$ up to 90% accuracy in a single stride
2017, Valada	The offline experiment used a deep Long-Short Term Memory (LSTM) based recurrent model that captures both the spatial and temporal dynamics of vehicle-terrain interaction sounds to classify mixed terrain-types	97.6% mean classification accuracy
2017, Hang Wu	Evaluates a deep learning framework for terrain visual classification against traditional (SVM and MLP) machine learning methods for 11 texture types from two image datasets	Mean classification score of 87% and 92% for feature learning and manual feature extraction methods respectively.

Table 2.3 Research summary part 3

Year, Author	Research Highlights	Remarks
2017, Dutta, Ayan Dasgupta, Prithviraj	Implements a multi-modality approach using Ensemble learning with weak classifier for acceleration, angular rate and roll-pitch yaw in identifying 5 terrain types	up to 63% better prediction accuracy compared to a support vector machine technique
2017, Zhao, Kai Dong, Mingming Gu, Liang	Implements a two stage manual feature selection method with a multi-classifier combination for five data sources from vibration and acoustics in identifying six terrain types	up to 10% accuracy increase when benchmarked with SVM and kNN classifiers
2018, Zhang, Lamei Chen, Zexi Zou, Bin Gao, Ye	Evaluates the classification performance of a 3D Convolutional Neural Network (CNN) against 2D CNN for a wide area containing 15 classes from high resolution aerial PolSAR images	Average accuracy of 95.29% versus 93.55% for 3D and 2D CNN respectively.

Table 2.4 Research summary part 4

Chapter Three

Methodology

In this section, an overview of the system hardware and software, including the wheeled robot setup, accelerometer hardware, data logging, and the data analysis are provided. Two surfaces, tiles and gravel were used in this experiment along with two approaches for classification which are Feature engineering and Feature learning. Due to limited knowledge during this implementation, a single modality approach, vibration, was employed to evaluate the effectiveness of Feature engineering and Feature learning in terrain classification. Thus, subsequent work will focus on integrating multi-modal (eg textures, acoustics) methods.

3.1 Wheeled Robot

3.1.1 Assembly

The experiment in this paper utilizes a 240Mhz dual-core 32-bit ESP32 MCU with built-in 2.4GHz Wi-Fi and Bluetooth support. It also includes peripherals such as PWM, SPI, UART and I2C. Here, an L298N motor driver is interfaced with the MCU to drive two DC motors forward while an accelerometer is connected via I2C to log acceleration and angular

velocity as the robot is moved forward. The logged data is sent to a registered ThingSpeak channel via an HTTP request where it can be analysed.

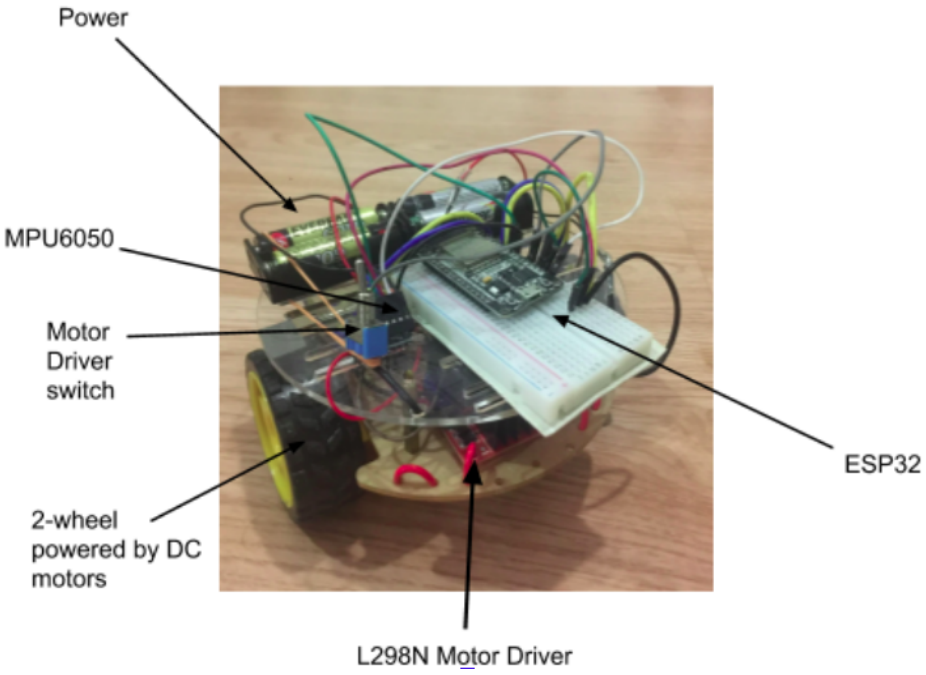

Figure 3.1 Two-wheeled robot

3.1.2 Experiment setup

The wheeled robot is first placed on the corresponding surface. A 6 second traversal time was employed in which 300 acceleration samples were collected in that period. It was found that this number of samples was not sufficient for training the classifiers effectively and thus a second run over the same traversal time was employed right after the first to obtain more data. This amounted to about 600 collected samples per surface. Figure 3.2 below shows the workflow of the setup and is explained in the following subsections.

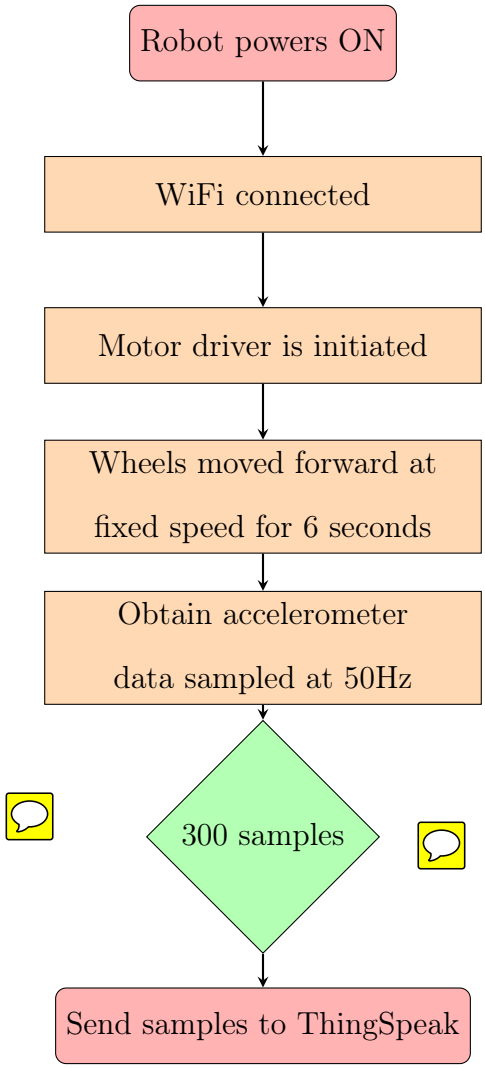

Figure 3.2 Flowchart for obtaining data

3.1.3 Robot movement

A two-wheeled robot was employed in this preliminary experiment and thus the only direction in which the robot was traversing is in the forward direction. The motor speed can be adjusted from a Pulse Width Modulation (PWM) from 0 to 255. For this experiment, a sufficient speed was obtained at a PWM of 225 and this value was constant throughout the experiment on both surfaces. Figures 3.3 AND 3.4 shows how the wheeled robot on the tile and gravel surfaces just before moving.

Figure 3.3 Tiles

Figure 3.4 Gravel

3.2 Acquisition of Sensor Data

The accelerometer used to gather data is an MPU-6050 which combines a 3-axis gyroscope and 3-axis accelerometer on the same silicon chip.

3.2.1 Sampling

From the MPU-6050's manufacturer datasheet, the maximum sampling rates for the Gyroscope and Accelerometer are 8000Hz and 1000Hz respectively. Thus, for the purposes of this experiment, the maximum sampling rate from the MPU-6050 is 1000Hz since the acceleration and angular velocity data are simultaneously logged by the MCU. Based on experiments conducted focusing on vibration based terrain classification, a variety of sampling rates were used. 76.25Hz from [48] and 64Hz from [50]. The most common sampling rate was found to be 50Hz as described in [4]. According to Nyquist Sampling Theorem, an analog signal waveform can be converted into digital by sampling the analog signal at equal time intervals. The sampling rate must be "equal to, or greater than, twice the highest frequency component in the analog signal.

$$f_s \geq f_{max} \tag{3.1}$$

It is demonstrated that for an application requiring sampling from their accelerometer and gyroscope, 99 percent of the frequency spectrum obtained from the Fast Fourier Transform is contained below 15kHz [29]. With the Nyquist criterion in mind, a sufficient value of 50Hz (50 samples per second) is used for this experiment to sample the MPU-6050.

3.2.2 Logging Data

Traditional methods of data logging incorporate an SD card to store sensor data. In this experiment, the built-in WiFi chip of the MCU was instead used to send logged data wirelessly and the ThingSpeak platform was used to aggregate and visualize live data streams from the MCU. There were two important points to take into consideration. The first takes into account the format of the sensor reading data expected from ThingSpeak and the second which deals with the usage of ThingSpeak's API's to facilitate the transfer of that data.

In each of the preliminary experiments using different types of surfaces, the average run time of the robot was 6 seconds. Thus, 300 samples containing accelerometer readings were obtained per run for a sampling frequency of 50Hz. The Bulk-Write JSON Data API was used to facilitate the transfer of data to the ThingSpeak platform over HTTP. A JSON formatting library was used to format sensor readings as the API expects a JSON object (JavaScript Object Notation) containing all the required data such as the API key, timestamps and the sensor readings. To ensure consistency when running the experiment on carpet and tile surfaces, a size of 300 elements corresponding to 300 samples was set as the maximum size for the JSON object. The data includes acceleration in the x, y, z direction and Gyroscope values in the x, y, z direction. Two runs were employed to obtain more readings and thus, 2 x 300 = 600 total vibration samples were obtained for each surface. These

samples were compiled in one CSV file for each surface type.

Time Series Visualisation

The sensor data containing acceleration values sent to ThingSpeak was exported as a CSV file to MATLAB where it is then processed. Figures 3.5, 3.6 displays a time series representation of the first few acceleration values for one run in the x-direction for gravel and tiles surfaces respectively.

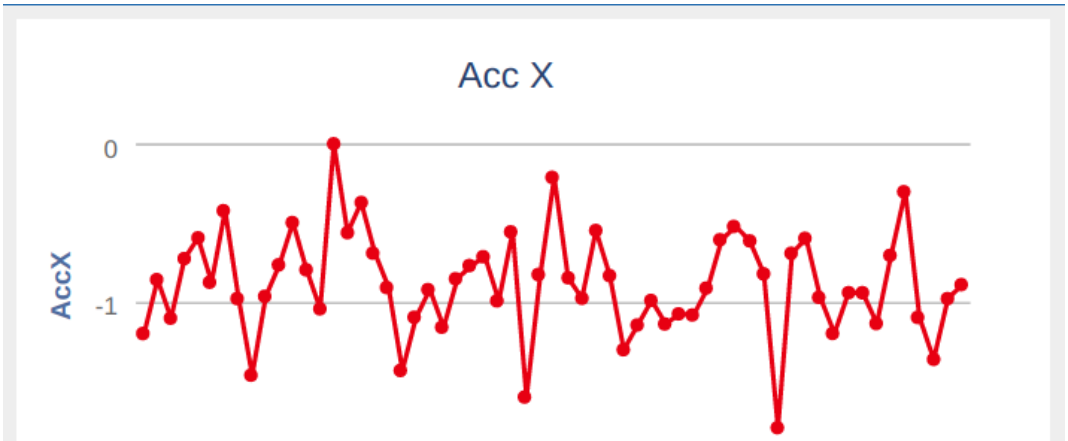

Figure 3.5 Time Series Visualisation (Gravel)

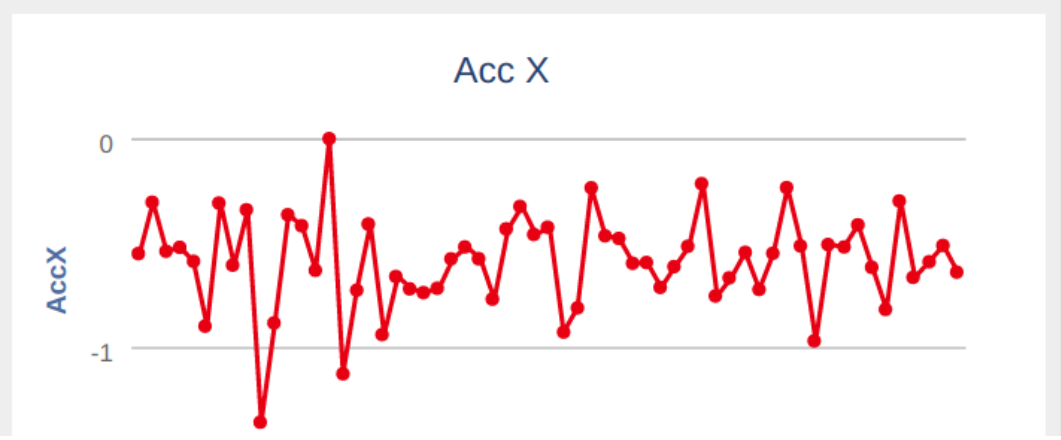

Figure 3.6 Time Series Visualisation (Tiles)

3.3 Feature Engineering Approach

This approach uses the vibration data obtained from the mounted accelerometer to generate handcrafted features. Figure 3.7 illustrates this approach and is explained in detail after.

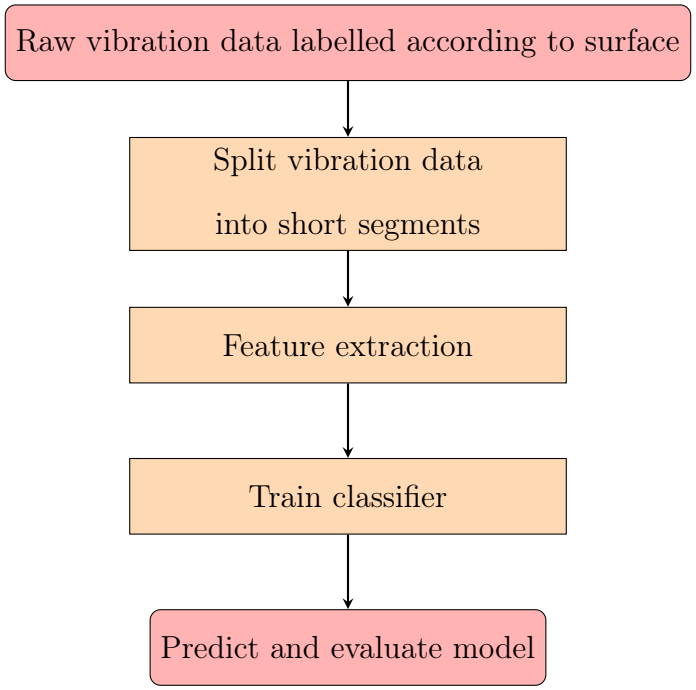

Figure 3.7 Flowchart for feature engineering

3.3.1 Segmenting raw vibration data

The 600 vibration samples for each surface obtained from the experiment was segmented into lengths of 10 with no overlap between segments. A length of 10 vibration samples constitutes a traversal time of 10 x 0.02 = 0.2 seconds (samples x sampling period). Thus, about 60 segments (for each surface) containing 10 samples of raw vibration data was obtained and is

used for feature extraction.

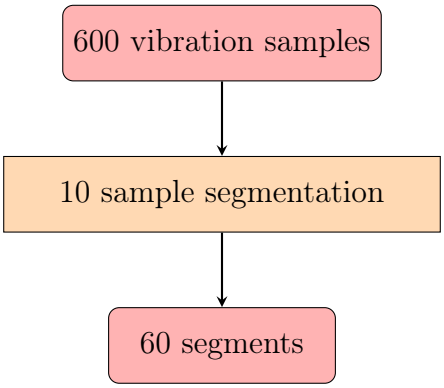

Figure 3.8 Segmentation

3.3.2 Feature extraction from segmented vibration data

Feature extraction is performed in the time domain and in the frequency domain. Time-domain vibration feature extraction aims to obtain a simple and compact representation of vibration signal. Here, the mean, standard deviation, norm, RMS, maximum and minimum are usually selected as statistical features [4]. In vibration-based damage detection, RMS, skewness and kurtosis are used since they have been proved to be useful for bearing fault detection [51].

Moreover, the frequency-domain representation of the vibration samples was obtained to be added to the overall feature vector. The most common tool for frequency-domain transformation is the Fast Fourier Transform (FFT), an efficient computing algorithm that is an extensive application in both vibration based terrain classification [48] and acoustics-based terrain classification [40]. Table 3.2 summarises the features employed in this experiment

from the time and frequency domains.

Considering the sample set $D = (v_1, v_2, \ldots, v_m)$, where m is the number of samples and for each sample $v_j = (v_{j,1}, v_{j,2}, \ldots, v_{j,n})$, where n is the number of acceleration values in a segment, the most features employed in this experiment is summarised in Table 3.3.2.

Domain	Features	Total
Time	mean, standard deviation, maximum, minimum, root mean square, power, skewness, kurtosis, mean absolute deviation, energy	10
Frequency	band power, maximum frequency, minimum frequency and mean frequency	4

Table 3.1 Features obtained from vibration data

Thus, for acceleration data in one direction, 14 features are generated. Considering axes from x, y, z, this generates a 42 sized feature vector.

3.3.3 Classification and results

Various classifiers have been employed for vibration based classification. Among them, Support Vector Machine is used the most widely. Other common classifiers such as k-nearest neighbor (kNN), decision tree (DT), Naïve Bayes (NB) and extreme learning machine (ELM) have also been applied [4]. Additionally, ensemble learning such as RF and AdaBoost, is currently a research hotspot in machine learning for its superior generalization performance [39].

In this experiment, an SVM classifier was trained on the generated features with 70% used for training and 30% used for validation. The *fitcsvm* MATLAB function was used to train the support vector machine (SVM) model for two-class (binary) classification on the moderate-dimensional feature data set. The *radial basis kernel* was set along with auto hyperparameter optimization yielding the following confusion matrix where classes 1, 2 corresponds to roads and tiles respectively. The implemented MATLAB code can be found in this repository.

Figure 3.9 Confusion matrix for SVM classifier on roads and tiles

3.4 Feature Learning Approach

The classification performance of feature-engineering approaches relies on handcrafted feature extraction heavily. In recent years, there has been a considerable effort on the development of end-to-end learning methods. Instead of manually extracting characteristic features, end-to-end learning method can learn the discriminative feature representation directly from raw data. The latter approach does not require too much prior knowledge of the problem or human expertise, and is advantageous in tasks where some high-level, abstract features from raw data are almost impossible to be developed manually. Usually, end-to-end learning method like deep neural network suffers from computationally intensive training process. However, once the network is trained, it can be directly assembled to the mobile robots, thus not computationally intensive. Figure 3.10 illustrates this approach and is explained in detail after.

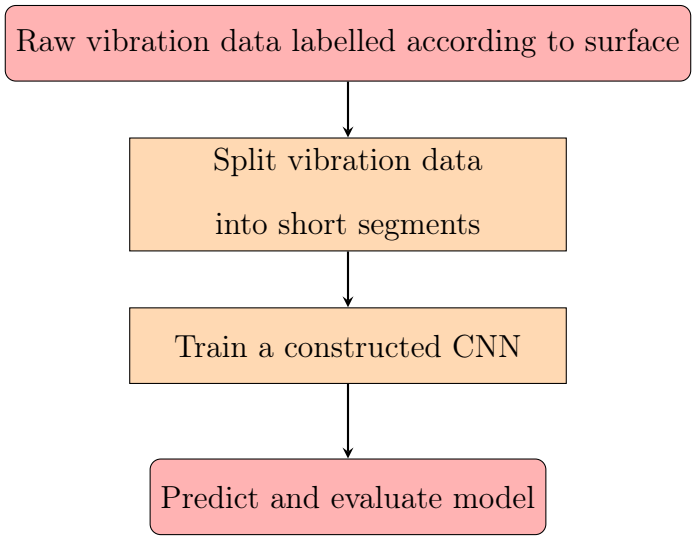

Figure 3.10 Flowchart for feature learning

3.4.1 Segmenting raw vibration data

Similar to the Feature Engineering approach in 3.4.1, the 600 vibration samples for each surface was segmented into lengths of 10. However, the only difference is that a 50% overlap was applied between successive segments.

3.4.2 Construct Convolutional Neural Network

Based on literature review, the CNN-based vibration terrain classification has rarely been investigated. Although, many existing CNN are often general-purpose, this initial experiment evaluates them with subsequent work focused on designing a dedicated neural network model by modifying and integrating CNN and LSTM. Such a neural network model can learn both spatial and temporal characteristics of the raw vibration signals.

The implemented CNN in this experiment consists of a stacked two layer convolutional neural network with an applied rectified linear unit (ReLU) as the nonlinear activation function. Next, two dropout layers were stacked to tackle overfitting issues with the final layer utilising the softmax activation function to interpret the output of the CNN as a probability. The Adam optimization technique was used during the compilation of the model which optimizes the CNN by leveraging the power of adaptive learning rates methods to find individual learning rates for each parameter. The development and evaluation of the CNN was done on a Python environment using Keras, Scikit Learn and Tensorflow. This was due to the fact that Curtin's MATLAB license does not include the Deep Learning toolbox needed to perform this part of the experiment. The code was implemented on a Jupyter notebook and can be found in this repository.

3.4.3 Classification and results

In this part of the experiment, the CNN was trained on the segmented vibration data with 70% used for training and 30% used for validation, the same for the Feature engineering approach. The learning curve for the model's accuracy and loss is plotted in Figures 3.11 and 3.12. The confusion matrix is found in 3.13.

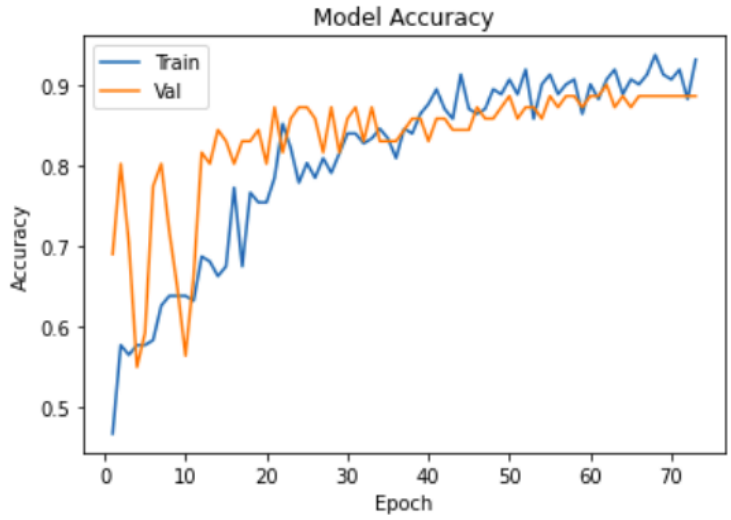

Figure 3.11 Learning curve for model accuracy

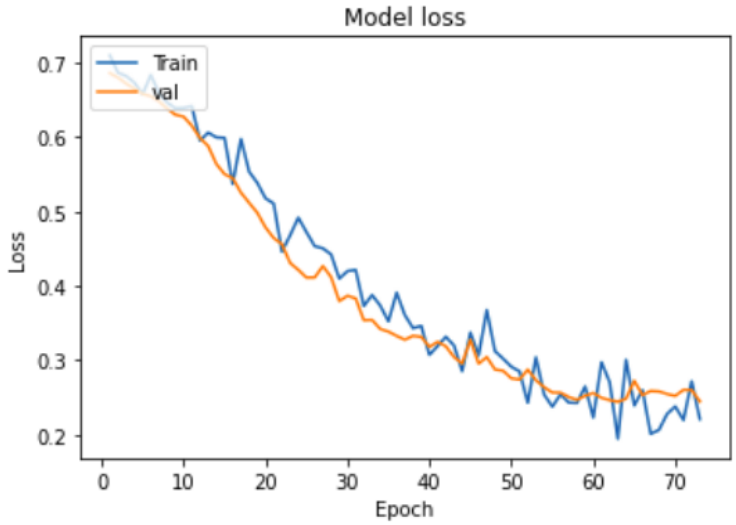

Figure 3.12 Learning curve for model loss

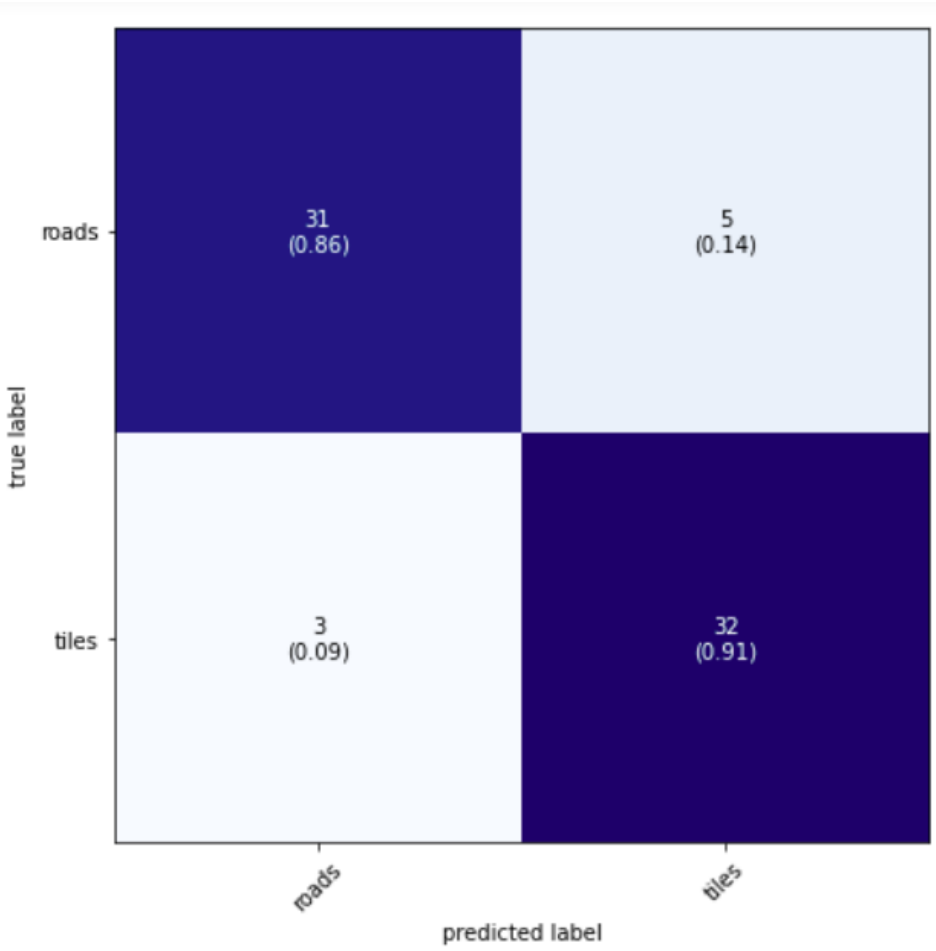

Figure 3.13 Confusion matrix for CNN on roads and tiles

3.5 Summary of preliminary results

Approach	Algorithm	Mean Accuracy
Feature engineering	Support Vector Machine (SVM)	91.67%
Feature extraction	Convolutional Neural Networks (CNN)	88.5%

Table 3.2 Summary of preliminary results

Chapter Four

Conclusion and Future works

In this progress report, the research objective and aim was first made relating to the relevance of terrain classification in autonomous mobile systems with an extensive comparative study of terrain classification approaches for justification. The aim of the research was to evaluate the effectiveness of feature engineering and feature learning methods in terms of computational cost and accuracy for multi-modal terrain classification. According to this research goal, one can find the most appropriate classification method according to their requirements.

This progress report achieves preliminary classification results on two surfaces which are tiles and gravel. An end to end embedded system was first built for vibration data collection utilising a mounted accelerometer on the mobile robot. Next, data preparation and processing was done on the obtained raw vibration signals which is to be used for two approaches in terrain classification. The feature approach relied on manual feature extraction of which the features are to be passed to a classifier. For 14 generated features, the trained Support Vector Machine (SVM) employed in this experiment achieves a mean 91.67% score. The second approach relied on a feature learning method using a Convolutional Neural Network (CNN) for classification. The trained CNN achieves a mean score of 88.5%.

A considerable amount of time was spent on implementing the end to end pipeline consisting of data collection from the mobile robot, data processing and preparation, understanding and constructing the SVM and CNN for classification. Thus, subsequent work to follow includes work that has not been accomplished in this progress report. This consists of multi-modal data collection (i.e texture, acoustics) on numerous other terrain types, investigation of varying segment lengths and comparison of classification accuracy and classification times of numerous other terrains with the the goal of validating and implementing the most suited approach in an actual mobile robot.

REFERENCES

[1] Hassani, I., Maalej, I., and Rekik, C. (2018). Robot Path Planning with Avoiding Obstacles in Known Environment Using Free Segments and Turning Points Algorithm. Mathematical Problems in Engineering, 2018.

[2] Wurm, K. M., Kümmerle, R., Stachniss, C., and Burgard, W. (2009). Improving robot navigation in structured outdoor environments. 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2009, November, 1217–1222.

[3] Rubio, F., Valero, F., & Llopis-Albert, C. (2019). A review of mobile robots: Concepts, methods, theoretical framework, and applications. International Journal of Advanced Robotic Systems, 16(2), 1–22.

[4] C. Weiss, N. Fechner, M. Stark, and A. Zell, "Comparison of different approaches to vibration-based terrain classification," in Proceedings of the in 3rd European Conference on Mobile Robots, pp. 7–12, Freiburg, Germany, September 2007

[5] H. Inotsume, K. Skonieczny, and D. S. Wettergreen, "Analysis of grouser performance to develop guidelines for design for planetary rovers," in Proceedings of the in Proceedings of the 12th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS 2014, pp. 1–9, Montreal, Canada, June, 2014.

[6] M. Ono, T. J. Fuchs, A. Steffy, M. Maimone, and J. Yen, "Risk-aware planetary rover operation: Autonomous terrain classification and path planning," in Proceedings of the 2015 IEEE Aerospace Conference, AERO 2015, pp. 1–10, Big Sky, Mont, USA, March 2015.

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

[8] Wolf, D. F., Sukhatme, G. S., Fox, D., Burgard, W. (2005). Autonomous terrain mapping and classification using hidden Markov models. Proceedings - IEEE International Conference on Robotics and Automation, 2005(January), 2026–2031.

[9] Lalonde, J. F., Vandapel, N., Huber, D. F., Hebert, M. (2006). Natural terrain classification using three-dimensional ladar data for ground robot mobility. Journal of Field Robotics, 23(10), 839–861.

[10] Brooks, C. A., Iagnemma, K. (2009). Visual detection of novel terrain via two-class classification. Proceedings of the ACM Symposium on Applied Computing, 1145–1150.

[11] Howard, A., Seraji, H. (2001). Vision-based terrain characterization and traversability assessment. Journal of Robotic Systems, 18(10), 577–587.

[12] Sofman, B., Bagnell, J. A., Stentz, A., Vandapel, N. (2006). Terrain Classification from Aerial Data to Support Ground Vehicle Navigation (52 cites). Robotics, 6.

[13] M. J. Matari´c. The Robotics Primer. The MIT Press, 1st edition, 2007.

[14] R. Negenborn. Robot localization and kalman filters. Master's thesis, Utrecht University, Netherlands, September 2003

[15] S. Thrun, W. Burgard, and D. Fox. Probabilistic Robotics. The MIT Press, September 2005.

[16] D. Fox. Markov Localization: A Probabilistic Framework for Mobile Robot Localization and Navigation. PhD thesis, Institute of Computer Science III, University of Bonn, Germany, December 1998

[17] N. Gordon, D. Salmond, and A. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. IEEE Proceedings for Radar and Signal Processing,

140(2):107–113, April 1993.

[18] D. G. Lowe. Object recognition from local scaleinvariant features. The Proceedings of the Seventh IEEE International Conference on Computer Vision, 2:1150–1157, 1999.

[19] D. Prasser and G. Wyeth. Probabilistic visual recognition of artificial landmarks for simultaneous localization and mapping. Proceedings of the 2003 IEEE International Conference on Robotics and Automation, 1:1291–296, September 2003.

[20] Weerakoon, T., Ishii, K., Nassiraei, A. A. F. (2015). An Artificial Potential Field Based Mobile Robot Navigation Method To Prevent From Deadlock. Journal of Artificial Intelligence and Soft Computing Research, 5(3), 189–203.

[21] Nacional, C. (2004). Relationship between Genetic Algorithms and Ant Colony Optimization Algorithms. Quality, 11(4), 1–16.

[22] SONG, K.-C., YAN, Y.-H., CHEN, W.-H., ZHANG, X. (2013). Research and Perspective on Local Binary Pattern. Acta Automatica Sinica, 39(6), 730–744.

[23] Khan, Y. N., Masselli, A., Zell, A. (2012). Visual terrain classification by flying robots. Proceedings - IEEE International Conference on Robotics and Automation, May 2014, 498–503.

[24] Anantrasirichai, N., Burn, J., Bull, D. (2015). Terrain Classification from Body-Mounted Cameras during Human Locomotion. IEEE Transactions on Cybernetics, 45(10), 2249–2260.

[25] Chetan, J., Krishna, M., Jawahar, C. V. (2010). Fast and spatially-smooth terrain classification using monocular camera. Proceedings - International Conference on Pattern Recognition, August 2010, 4060–4063.

[26] Aldavert, D., Rusiñol, M., Toledo, R., Lladós, J. (2015). A study of Bag-of-Visual-Words representations for handwritten keyword spotting. International Journal on Document Analysis and Recognition, 18(3), 223–234.

[27] Van Gemert, J. C., Veenman, C. J., Smeulders, A. W. M., Geusebroek, J. M. (2010). Visual word ambiguity. IEEE Transactions on Pattern Analysis and Machine Intelligence, 32(7), 1271–1283.

[28] Wu, H., Liu, B., Su, W., Chen, Z., Zhang, W., Ren, X., Sun, J. (2017). Optimum Pipeline for Visual Terrain Classification Using Improved Bag of Visual Words and Fusion Methods. Journal of Sensors, 2017.

[29] Filitchkin, P., Byl, K. (2012). Feature-based terrain classification for LittleDog. IEEE International Conference on Intelligent Robots and Systems, 2, 1387–1392.

[30] Shallue, C., Vanderburg, A. (2018). Identifying exoplanets with deep learning: A five planet resonant chain around kepler-80 and an eighth planet around kepler90. Astronomical Journal, [In Press].

[31] Schroff, F., Kalenichenko, D., Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In IEEE Conference on Computer Vision and Pattern Recognition. Boston, MA, USA: IEEE.

[32] Zeiler, M., Fergus, R. (2013). Stochastic pooling for regularization of deep convolutional neural networks. In International Conference on Learning Representations. Scottsdale, AZ, USA.

[33] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15, 1929 – 1958.

[34] Gonzalez, R., Iagnemma, K. (2018). DeepTerramechanics: Terrain Classification and Slip Estimation for Ground Robots via Deep Learning. http://arxiv.org/abs/1806.07379

[35] Zeiler, M., Fergus, R. (2014). Visualizing and understanding convolutional networks. In European Conference on Computer Vision (pp. 818 – 833). Zurich, Switzerland.

[36] Zhang, L., Chen, Z., Zou, B., Gao, Y. (2018). Polarimetric SAR terrain classification using 3D convolutional neural network. International Geoscience and Remote Sensing Symposium (IGARSS), 2018-July, 4551–5454.

[37] Wang, Z., Sun, Y., Shen, Q., Cao, L. (2019). Dilated 3d convolutional neural networks for brain mri data classification. IEEE Access, 7, 134388–134398.

[38] Libby J., Stentz A.J. Using sound to classify vehicle-terrain interactions in outdoor environments; Proceedings of the 2012 IEEE International Conference on Robotics and Automation; St. Paul, MN, USA. 14–18 May 2012; pp. 3559–3566.

[39] Valada A., Burgard W. Deep spatiotemporal models for robust proprioceptive terrain classification. Int. J. Robot. Res. 2017;36:1521–1539. doi: 10.1177/0278364917727062.

[40] J. Christe and N. Kottege. Acoustics based terrain classification for legged robots. In Proceedings of the IEEE International Conference on Robotics and Automation, 2016.

[41] Hoepflinger M.A., Remy C.D., Hutter M., Spinello L., Siegwart R. Haptic terrain classification for legged robots; Proceedings of the 2010 IEEE International Conference on Robotics and Automation; Anchorage, Alaska. 3–8 May 2010; pp. 2828–2833.

[42] Chi, C., Sun, X., Xue, N., Li, T., Liu, C. (2018). Recent progress in technologies for tactile sensors. Sensors (Switzerland), 18(4). https://doi.org/10.3390/s18040948

[43] Al-Handarish, Y., Omisore, O. M., Igbe, T., Han, S., Li, H., Du, W., Zhang, J., Wang,

L. (2020). A Survey of Tactile-Sensing Systems and Their Applications in Biomedical Engineering. Advances in Materials Science and Engineering, 2020.

[44] Hoffmann M., Štěpánová K., Reinstein M. The effect of motor action and different sensory modalities on terrain classification in a quadruped robot running with multiple gaits. Robot. Auton. Syst. 2014;62:1790–1798.

[45] E. Ugur, E. Oztop, E. Sahin, Goal emulation and planning in perceptual space using learned affordances, Robot. Auton. Syst. 59 (7–8) (2011) 580–595.

[46] M. Lungarella, O. Sporns, Mapping informat ion flow in sensorimotor networks, PLoS Comput. Biol. 2 (2006) 1301–1312.

[47] Wu, X. A., Huh, T. M., Mukherjee, R., Cutkosky, M. (2016). Integrated Ground Reaction Force Sensing and Terrain Classification for Small Legged Robots. IEEE Robotics and Automation Letters, 1(2), 1125–1132.

[48] Brooks C.A., Iagnemma K. Vibration-based terrain classification for planetary exploration rovers. IEEE Trans. Robot. 2005;21:1185–1191.

[49] Bermudez F.L.G., Julian R.C., Haldane D.W., Abbeel P., Fearing R.S. Performance analysis and terrain classification for a legged robot over rough terrain; Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems; Algarve, Portugal. 7–12 October 2012; pp. 513–519.

[50] Otte S., Weiss C., Scherer T., Zell A. Recurrent Neural Networks for fast and robust vibration-based ground classification on mobile robots; Proceedings of the IEEE International Conference on Robotics and Automation; Stockholm, Sweden. 6–20 May 2016; pp. 5603–5608.

[51] Abdeljaber O., Avci O., Kiranyaz S., Gabbouj M., Inman D.J. Real-time

vibration-based structural damage detection using one-dimensional convolutional neural networks. J. Sound Vib. 2017;388:154–170. doi: 10.1016/j.jsv.2016.10.043.

[52] Otsu K., Ono M., Fuchs T.J., Baldwin I., Kubota T. Autonomous terrain classification with co-and self-training approach. IEEE Robot. Autom. Lett. 2016;1:814–819. doi: 10.1109/LRA.2016.2525040.

[53] Halatci, I., Brooks, C. A., Iagnemma, K. (2007). Terrain classification and classifier fusion for planetary exploration rovers. IEEE Aerospace Conference Proceedings.

[54] Zhao K., Dong M., Gu L. A New Terrain Classification Framework Using Proprioceptive Sensors for Mobile Robots. Math. Probl. Eng. 2017;2017:3938502. doi: 10.1155/2017/3938502. [CrossRef] [Google Scholar] [Ref list] [Google Scholar] [Ref list]