Deep Learning for Terrain Surface Classification: Vibration-based Approach

Marcos Concon[a], W. K. Wong[a], Filbert H. Juwono[a] and Catur Apriono[b]

[a]Curtin University Malaysia, CDT 250, Miri 98009, Sarawak, Malaysia

[b]University of Indonesia, Kampus Baru UI, Depok, West Java 16424, Indonesia

Abstract

As robots become more pervasive in the service sector, control in dynamic environment has become an important element in optimising the deployment of mobile robots. A mobile robot should be knowledgeable not only of the barriers, but also of the surface on which the robot navigates to estimate slippage and adaptive control. We note that various terrains/surfaces have different characteristics, which can directly influence the handling, driving, efficiency, and stability of the robot vehicle. Knowledge of the terrain can provide valuable information for establishing effective and secure navigation strategies. We built a mobile robot prototype equipped by Inertial Measurement Unit (IMU) to obtain the terrain data and applied deep learning models to classify the terrain using the data. Three deep learning configurations have been proposed in this paper, i.e. long short-term memory (LSTM), 1D convolutional network (1D CNN), and convolutional neural network-long short-term memory network (CNN-LSTM). The deep learning architectures were trained and evaluated based on the data collected from five different surfaces. It is shown that the CNN-LSTM performs the best with an F1 score of 98.49%. The other two networks also generalize relatively well with the unseen vibration sequences with F1 scores of 97.47% and 95.98% for the 1D CNN and LSTM, respectively. Finally, we investigate the effect of varying input sequence to find the optimal length, so that we are able to obtain the highest accuracy and generalization of the deep learning networks.

Keywords

LTSM, CNN, terrain classification,

1. Introduction

Intelligent robotics have seen rapid advancement in their scope of operations such as in military reconnaissance in hostile environments [1], unmanned surveillance for disaster management [2], and telemedicine robot used for examining remote patients [3], and in factories. It is necessary for a robot to acquire a clear understanding of its current environment in order to successfully manoeuvre and accomplish its planned operation, while preventing any damage to itself and creating hazards to others. As service robots have achieved broad adoption in the above-mentioned industries, precise navigation and surrounding awareness have become crucial issues to improve the capacity of the device to deploy. An significant consideration for the robot's efficient navigation is the motion control algorithm, based on the type of terrain being travelled. Thus, a detailed classification of the type of terrain is required for the robot to adapt its speed of navigation and the parameters of route planning, which depend on the characteristics of the terrain.

In this paper, we focus on the area of terrain mapping using Inertial Measurement Unit (IMU) sensors. In order to map the readings to the respective terrain labels, we present and evaluate three types of deep learning frameworks: long short-term memory (LSTM), one dimensional convolutional neural network (1D-CNN), and the CNN-LSTM architectures. Both LSTM and CNN have been extensively used in the literature; however, the applications of utilizing both frameworks in a unified structure have been lacking. This paper aims to leverage the temporal and spatial advantages towards the vibration-based terrain classification.

It is worth noting that deep learning can be used for tasks where it is almost impossible to execute a raw data engineering function manually. Despite being highly 'blackbox', the end-to-end deep learning approach is suitable for automatically extracting useful features in complex non-linear classification tasks. Therefore, deep learning method can be implemented to obtain more reliable results to recognize the surrounding environment of the robot, thereby enhancing the robot's adaptive controls and mobility.

2. Related Works

The problems of adaptive control in mobile robots have been constantly researched. The challenges presents various opportunities for researchers to develop methods in predicting the dynamic changes in the environment. In

ISIC 2021: International Semantic Intelligence Conference, February 25–27, 2021, New Delhi, India

marcos@respiree.com (M. Concon); weikitt.w@curtin.edu.my (W.K. Wong); filbert@ieee.org (F.H. Juwono); catur@eng.ui.ac.id (C. Apriono)

0000-0001-6212-6096 (W.K. Wong); 0000-0002-2596-8101 (F.H. Juwono); 0000-0002-7843-6352 (C. Apriono)

CEUR Workshop Proceedings (CEUR-WS.org)

[4], the authors investigated the use of kinematics-based analytic for wheel slippage calculation. The results were validated using collected data on a mobile platform. Similar work was found in [5] where the authors applied rolling resistance torque without using any additional sensors. Rolling resistance torque in multiple terrains can be acquired by reaction torque observer. Proposed concept was verified by using a differential drive mobile robot. In [6], wheel slips were estimated based on the odometric data. The collected data were analyzed using two different approaches, which were instantaneous estimator and temporal window approach. Results showed that temporal window approach yielded a better result.

In [7], researchers presented a solution of using laser-based point cloud generation to detect robot traversal surface. The researchers explored several terrains including carpet, coated asphalt, and asphalt. The solution was highly precise with high computation cost as it generated point clouds which needed to be further processed digitally. The authors stated that there were opportunities to further investigate how a mobile robotic platform could provide reliable and accurate surface prediction of the terrain for improving the navigation with prior knowledge of the surface. These research works have shown that there is strong motivation in investigating methods to enable service robots to have perception on the terrain and traversal surface.

Several sensing methodologies have been developed to tackle the problem of terrain classification. The methodology is typically categorized into two main groups: vision-based and reaction-based techniques. Traditional visual feature engineering approaches include the scale-invariant feature transform (SIFT) [8], speeded-up robust features (SURF) [9], and the bag of visual words (BOVW) [10], among many others. These algorithms pass the useful features of the images obtained from light detection and ranging (lidar) or stereo camera to a classifier to be trained and classified. In [11], raw grayscale terrain images were trained using deep convolutional network and the accuracy was 6% less than the support vector machine (SVM) classifier used jointly with the histogram of gradients (HOG) feature extractor.

While vision-based approaches are useful because of their high accuracy, they are vulnerable to distortion caused by lighting changes and other factors such as the realization of the surface's physical properties (e.g. material type and degree of hardness) [12]. Reaction-based techniques, on the other hand, utilize sensor measurements to obtain either the acoustics, haptics, or the vibration profiles for the classification. Acoustic-based classification relies on the use of microphone to record the sound of signal generated between the robot and terrain during traversal. Noise removal and smoothing techniques are necessary in traditional acoustic-based classification to achieve satisfactory results. This is due to some factors such as environmental noise and the robot's internal motor noise as described in [13]. A deep learning approach was applied in [14] where a CNN was developed and trained using the short time Fourier transform (STFT) spectograms extracted from the raw terrain audio signals. It was demonstrated that the network was robust even when the terrain audio signal was corrupted with the white Gaussian noise.

Haptic-based classification uses ground contact forces between a legged robot and terrain to describe different terrain properties. Typically, features such as the robot's stride frequency, peak and average motor torque in a single stride are used to train an SVM classifier [15]. In [16], a 1-dimensional CNN and an RNN architecture were implemented and evaluated when raw force/torque signals from a hexapod robot were passed to them. There was a significant improvement of about 15% in classification accuracy when compared to the SVM method with a Gaussian kernel.

The last reaction-based technique is based on the vibration characteristics of the terrain. It was first suggested in [17] where the vibration signal was measured using an accelerometer during the robot's traversal. In terms of performance, SVM has proven to be the best when trained on hand crafted time domain features such as skewness, impulse factor, and root mean square (RMS), along with frequency-domain features from the discrete Fourier transform (DFT). Experiments using a CNN for vibrational wheel slip estimation in ground robotics was carried out in [11]. The wheel torque, vertical acceleration and degree of pitch were used to train the classifier. The difference of 10% for classification was obtained before and after filtering the input data for the CNN which reinforces the generality of deep learning frameworks in extracting meaningful information directly from raw input vibration data.

3. Methodology

The mobile robot used in this research is a two wheel differential drive with an attached 6-axis accelerometer that measures six vibrational terrain signatures. The setup is shown in Fig. 2. The form is very similar to the conventional indoor service robots such as robotic vaccuum cleaner. The vibration characteristics are all dependent on the terrain's texture/material and robot movement. This study primarily aims to address terrain classification by utilising raw time-series vibration data as input to three implemented deep learning frameworks: LSTM, CNN, and a CNN-LSTM architectures. An overview of the experiment workflow is given in Fig. 1.

The data set used in this study contains a total of 24000 samples distributed evenly from five different terrain sources. The six features includes the lateral, longitu-

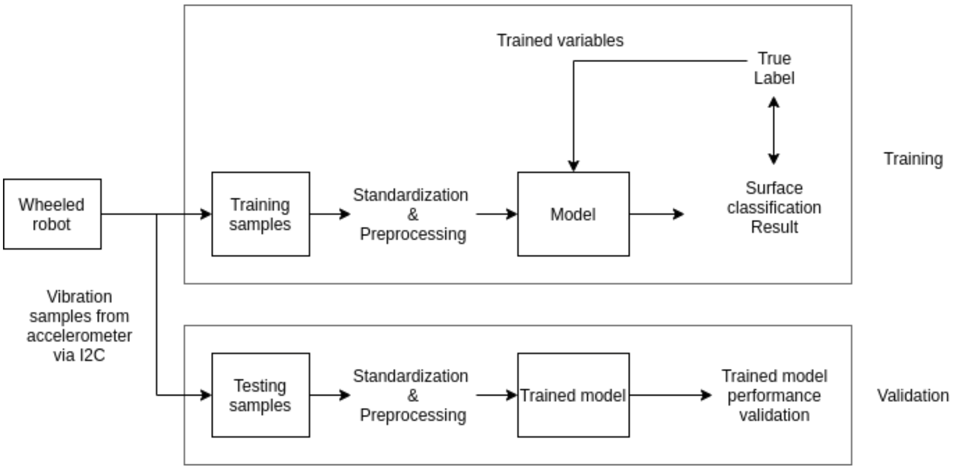
Trained variables
True Label
Training
Wheeled robot
Training samples
Standardization & Preprocessing
Model
Surface classification Result
Vibration samples from accelerometer via I2C
Testing samples
Standardization & Preprocessing
Trained model
Trained model performance validation
Validation

Figure 1: Experiment workflow.

dinal, and vertical accelerations and angular velocities $(a_x, a_y, a_z, g_x, g_y, g_z)$ of the traversing robot. The setup is shown in Fig. 2. Fig. 3 illustrates the five different vibration signals corresponding to the surface type. The vibration samples were collected via I2C using IMU unit MPU-6050 containing both an accelerometer and gyroscope integrated in a single chip. The controlled conditions for the wheeled robot are: 50 Hz sampling rate, 1.6 minutes traversal time per surface, and circular motion of the robot.

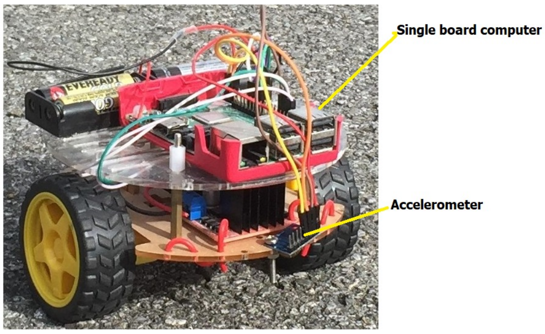

Figure 2: Experimental Setup.

Vibration samples must be converted into an appropriate format before entering the neural networks. Also, as the measurements contain multiple units, the range of vibration samples must be normalised to a mean of zero and a variance of one. The equation for normalization is given by

$$s_i, = \frac{x_i - \mu}{\sigma}, \qquad (1)$$

where i is the index of the element from the vibration sequence, μ is the average, and σ is the standard deviation. The vibration samples were then segmented into fixed windows of 1.5 seconds (75 samples). An overlap rate of 20% was applied between two consecutive 1.5 second segments to conserve the temporal dependencies between the time steps in the vibration sequence. One-hot encoding was then performed to map the different labelled surfaces numerically. Lastly, the vibration data set was split into training, validation, and testing sets to allow the neural networks to generalize with the unseen vibration characteristics. These data set partition were set to be 70%, 15% and 15% for training, validation, and testing, respectively.

3.1. Implementation

LSTM is a type of recurrent neural network (RNN) that is typically used for sequence prediction. In particular, LSTM solves the issues of the disappearing gradient present in the RNNs while allowing the long-term temporal dynamics of the series to be exploited. In contrast, CNNs have been commonly used for 2D problems (e.g. image classification task); however, it can be modified to classify the 1D vibrational problem. The dimensionality of the convolutional layers is reduced to match the model's 1D input.

The CNN-LSTM model leverages the robustness of CNN in extracting spatial features and LSTM in exploiting the temporal dependencies of the vibration sequence. In this paper, the time-series vibration is downsampled by the 1D CNN to extract the higher level features. This can be considered as the pre-processing step which allows the LSTM to interpret the features extracted at each block of the sequence. The concept is illustrated in Fig. 4.

The three models were built and trained using a Ten-

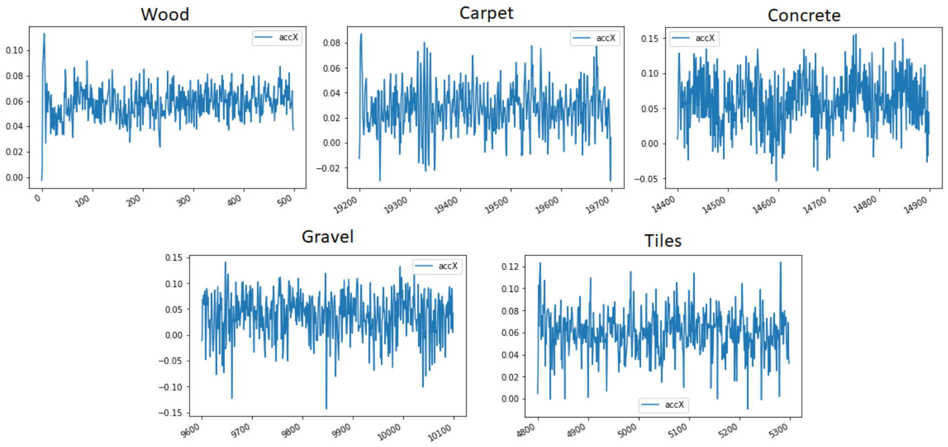

Figure 3: The five different terrain vibration signals.

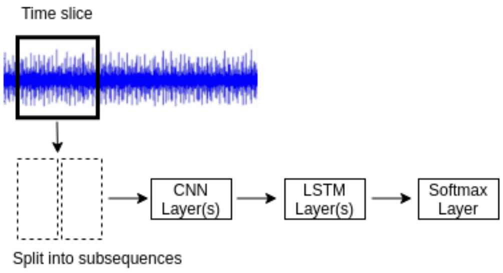

Figure 4: Time slice processing for CNN-LSTM

Table 1
Overview of the architecture used in this study

	Layer	Output shape
LSTM	LSTM (20 units)	(75, 20)
	Dropout (25%)	(75, 20)
	LSTM (70 units)	70
	Dropout (40%)	70
	Dense	112
	Dense	5
1D CNN	Conv1D (80@6×1)	(70, 80)
	Dropout (50%)	(70, 80)
	Conv1D (128@6×1)	(65, 128)
	Dropout(50%)	(65, 128)
	Max pooling	(32, 128)
	Flatten	4096
	Dense	96
	Dense	5
CNN-LSTM	Conv1D (96@6×1)	(3, 20, 96)
	Conv1D (48@6×1)	(3, 15, 48)
	Dropout (30%)	(3, 15, 48)
	Max pooling	(3, 7, 48)
	Flatten	(3, 336)
	LSTM (20 units)	60
	Dropout (20%)	60
	Dense	96
	Dense	5

sorflow backend with the Keras API. A detailed overview of the three models was summarized in Table 1. The hyperband algorithm was used to select the hyperparameters allowing for the best balance between training time and accuracy. The learning rate, batch size, and number of epochs were set at 0.001, 64, and 30, respectively. Additionally, early stopping regularization was implemented to avoid overfitting during model training. Further, the Adam optimization algorithm based on the Stochastic gradient descent was used as the optimizer.

The implemented CNN-LSTM architecture is shown in Fig. 5. For both the LSTM and 1D CNN networks, the data length of a vibration training sample was a flat vector of 75 time steps. In a stacked LSTM network, the input sequence to the first LSTM layer returns a shape of (timestep, unit) to be passed on to the next layer. The output from the last LSTM layer returns only the unit. For the 1D CNN, the input shape to the network is represented as (timestep, features). In the case for the CNN-LSTM network, a time distributed wrapper is first used before the LSTM layers to allow the input vibration signal to retain its temporal representation during the convolution process. The time distributed layer expects a 3D input and so the input sequence was reshaped from 75u time steps into 3 subsequences of 25 time steps. The convolutional layer used the ReLU activation and consisted of a 6 × 1 kernel that moves across in one dimension during the convolution operation.

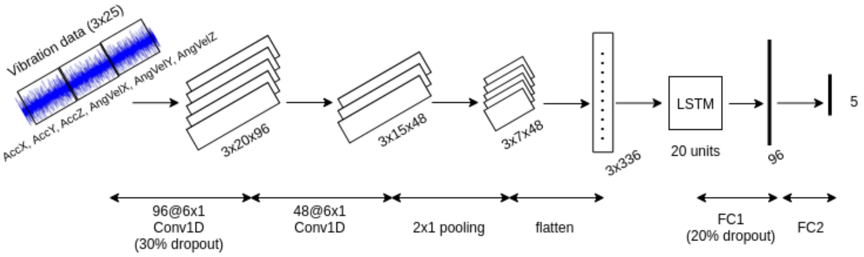

Figure 5: Implemented CNN-LSTM model for vibration-based terrain classification

The dropout layers were then added to tackle overfitting issues by arbitrarily setting a fraction rate of input units to zero. The pooling layer was added to reduce the spatial size of the output representation into half. Note that both the dropout and pooling layer allows for faster training time due to the reduced parameter size. The flatten layer was used to transform the input from the previous layers as input to the LSTM layer where the temporal characteristics of the vibration sequence were extracted. Lastly, the fully connected layers with the softmax activation function was used to structure the outputs of the previous layer for the final classification task. In this experiment, the categorical cross-entropy loss function was used to address the 5-class terrain classification problem.

4. Results

The confusion matrix of the three models are depicted in Fig. 6. From the confusion matrix, we can calculate the F1 score, the precision P_r, and the recall, R_c. The F1 score, which is calculated using P_r and R_c, has been commonly used to analyze the performance of the models. We used macro-averaging technique to expand these benchmarks towards multi-class terrain classification. The equations for the precision, recall, and F1 score, respectively are given by

$$P_r = \frac{T_P}{T_P + F_P}, \tag{2}$$

$$R_c = \frac{T_P}{T_P + F_N}, \tag{3}$$

$$F1 = \frac{2P_R R_C}{P_R + R_C}, \tag{4}$$

where T_P shows the outcome where the model correctly classifies the positive class, F_P is the outcome where the model incorrectly classifies the positive class, T_N is the outcome where the model correctly classifies the negative class, and F_N is the outcome where the model incorrectly classifies the negative class.

Fig. 7 illustrates the performance of the models on the 5-class vibration test data set. In the worst case, about 7% (average) of the wood class was mistakenly classified as tiles across the three models. Overall, it can be seen that the three models exhibited good performance and generalized well with the unseen data. Further, the CNN-LSTM architecture has the best performance with F1 score of 98.49% (average). The 1D CNN follows at the second place with F1 score of 97.49% (average). We note that the slight improvement of the CNN-LSTM model compared to the 1D CNN may suggest that the temporal characteristics of the LSTM is less important than the feature generation capability of the CNN-LSTM. Table 4 summarizes the overall performance of the three models.

Table 2
Average Precision, Recall, and F1 scores (Based on testing data)

Model	Precision	Recall	F1 Score
LSTM	96.23%	96.00%	95.98%
1D CNN	97.53%	97.50%	97.47%
CNN-LSTM	98.60%	98.50%	98.49%

One factor influencing the performance of the models is the sequence length of the input vibration. To further validate the performance of the three models, we analyzed the F1 score with varying segment lengths as shown in Fig. 8. It can be seen that a longer sequence length results in a better accuracy with the cost of performance saturation at a certain length. It can be shown that the average F1 score rises as the duration of the vibration series increases from 30 to 60 samples but decreases afterwards. This may be caused by the lack of the training data after the segmentation process of the given length. Therefore, we can consider an optimal sequence length of 75 (1.5 seconds) for obtaining high accuracy and generalization of the models. Furthermore, the proposed CNN-LSTM architecture slightly outperformed the CNN and LSTM models across the varying segment lengths.

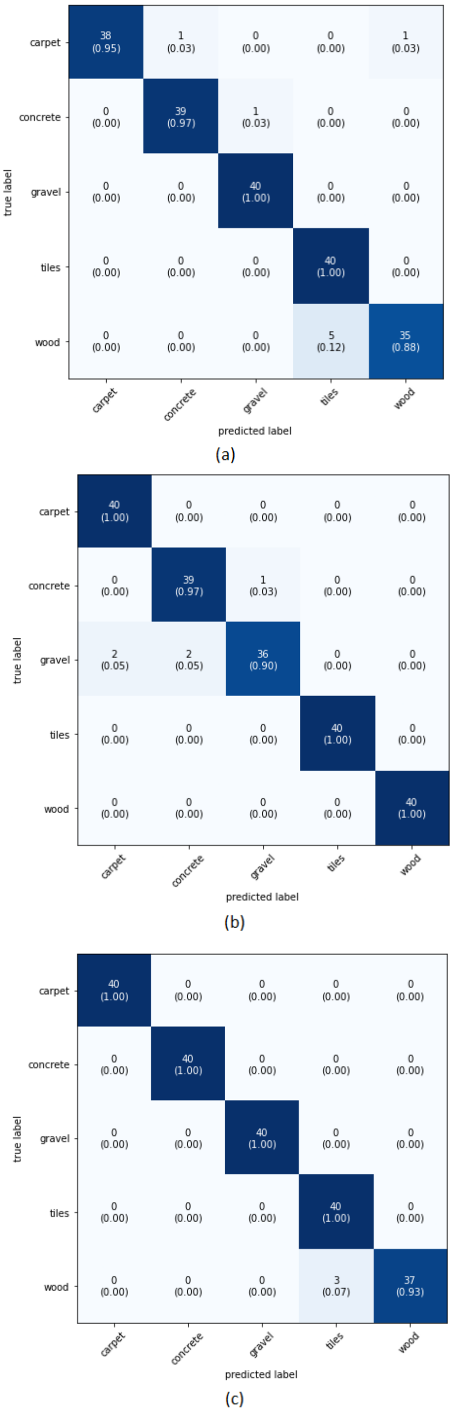

Figure 6: Confusion matrices of (a) LSTM, (b) CNN, and (c) CNN-LSTM on the vibration test dataset.

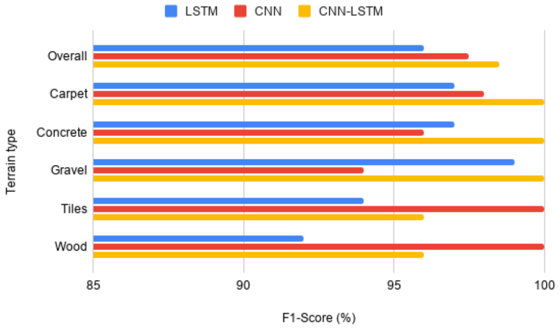

Figure 7: F1-scores for the three architectures

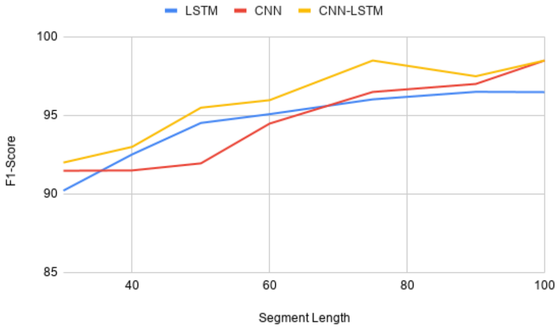

Figure 8: Average F1 Score at varying segment length

5. Conclusion and Future Work

In this paper, we have demonstrated the application of IMU-based surface classification task. We have compared three candidates for classifying the IMU data, i.e. LSTM, 1D CNN, and a combination of CNN and LSTM. By comparing the results, CNN-LSTM provided the best results (F1 score of 98.49%). However, we can further observe that the 1D CNN presented favorable results although slightly lower than the CNN-LSTM. The results suggest that 1D CNN is able to map the classification better when compared to the LSTM on standalone basis. CNN and LSTM works on different principle in which the latter is based on the temporal dynamics of the data. On the other hand, 1D CNN is based on static convolution, similar to the 2D counterparts. This implies that there is a clear static pattern when the IMU data enabling well defined mapping to their respective classes.

The results, despite counter intuitive, may prompt further research in the this direction. With the growing of edge computing and capacity of embedded system, enabling robots to recognize surface would enable further applications for indoor or industrial applications.

Reducing the complexity of the machine learning models to further benefit in terms of computation reduction is required. This is possible given that there is a clear static pattern demonstrated from the results using 1D CNN.

Acknowledgments

This research was supported by Ministry of Research and Technology/National Agency for Research and Innovation, Republic of Indonesia through Penelitian Dasar Unggulan Perguruan Tinggi (PDUPT) Grant, contract number: NKB-2838/UN2.RST/HKP.05.00/2020, year 2020.

References

[1] J. G. Bellingham, K. Rajan, Robotics in remote and hostile environments, Science 318 (2007) 1098–1102.

[2] V. Jorge, R. Granada, R. Maidana, D. Jurak, G. Heck, A. Negreiros, D. dos Santos, L. Gonçalves, A. Amory, A survey on unmanned surface vehicles for disaster robotics: Main challenges and directions, Sensors 19 (2019) 702. URL: http://dx.doi.org/10.3390/s19030702. doi:10.3390/s19030702.

[3] K. K. Chung, K. W. Grathwohl, R. K. Poropatich, S. E. Wolf, J. B. Holcomb, Robotic telepresence: past, present, and future, Journal of cardiothoracic and vascular anesthesia 21 (2007) 593—596.

[4] R. Chaichaowarat, W. Wannasuphoprasit, Wheel slip angle estimation of a planar mobile platform, in: 2019 First International Symposium on Instrumentation, Control, Artificial Intelligence, and Robotics (ICA-SYMP), 2019, pp. 163–166. doi:10.1109/ICA-SYMP.2019.8646198.

[5] S. D. A. P. Senadheera, A. M. H. S. Abeykoon, Sensorless terrain estimation for a wheeled mobile robot, in: 2017 IEEE International Conference on Industrial and Information Systems (ICIIS), 2017, pp. 1–6. doi:10.1109/ICIINFS.2017.8300422.

[6] D. Masha, M. Burke, b. Twala, Slip estimation methods for proprioceptive terrain classification using tracked mobile robots, in: International Conference (PRASA-RobMech), 2017, pp. 150–152.

[7] S. Wilson, J. Potgieter, K. Arif, Floor surface mapping using mobile robot and 2d laser scanner, in: 2017 24th International Conference on Mechatronics and Machine Vision in Practice (M2VIP), 2017, pp. 1–6. doi:10.1109/M2VIP.2017.8211508.

[8] S. Zenker, E. E. Aksoy, D. Goldschmidt, F. Wörgötter, P. Manoonpong, Visual terrain classification for selecting energy efficient gaits of a hexapod robot, in: 2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2013, pp. 577–584. doi:10.1109/AIM.2013.6584154.

[9] Seung-Youn Lee, D. Kwak, A terrain classification method for ugv autonomous navigation based on surf, in: 2011 8th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), 2011, pp. 303–306. doi:10.1109/URAI.2011.6145981.

[10] H. Wu, B. Liu, W. Su, Z. Chen, W. Zhang, X. Ren, J. Sun, Optimum pipeline for visual terrain classification using improved bag of visual words and fusion methods, Journal of Sensors 2017 (2017).

[11] R. González, K. Iagnemma, Deepterramechanics: Terrain classification and slip estimation for ground robots via deep learning, CoRR abs/1806.07379 (2018). URL: http://arxiv.org/abs/1806.07379.

[12] P. Roy, S. Ghosh, S. Bhattacharya, U. Pal, Effects of degradations on deep neural network architectures, CoRR abs/1807.10108 (2018). URL: http://arxiv.org/abs/1807.10108. arXiv:1807.10108.

[13] J. Libby, A. J. Stentz, Using sound to classify vehicle-terrain interactions in outdoor environments, in: 2012 IEEE International Conference on Robotics and Automation, 2012, pp. 3559–3566. doi:10.1109/ICRA.2012.6225357.

[14] A. Valada, L. Spinello, W. Burgard, Deep feature learning for acoustics-based terrain classification, in: International Symposium on Robotics Research (ISRR), 2015.

[15] X. A. Wu, T. M. Huh, R. Mukherjee, M. Cutkosky, Integrated ground reaction force sensing and terrain classification for small legged robots, IEEE Robotics and Automation Letters 1 (2016) 1125–1132. doi:10.1109/LRA.2016.2524073.

[16] J. Bednarek, M. Bednarek, L. Wellhausen, M. Hutter, K. Walas, What am i touching? learning to classify terrain via haptic sensing, in: 2019 International Conference on Robotics and Automation (ICRA), 2019, pp. 7187–7193. doi:10.1109/ICRA.2019.8794478.

[17] C. A. Brooks, K. Iagnemma, Vibration-based terrain classification for planetary exploration rovers, IEEE Transactions on Robotics 21 (2005) 1185–1191. doi:10.1109/TRO.2005.855994.