AI-Enabled Predictive Maintenance Framework for Autonomous Mobile Cleaning Robots

Sathian Pookkuttath, Mohan Rajesh Elara, Vinu Sivanantham and Balakrishnan Ramalingam

Abstract: Vibration is an indicator of performance degradation or operational safety issues of mobile cleaning robots. Therefore, predicting the source of vibration at an early stage will help to avoid functional losses and hazardous operational environments. This work presents an artificial intelligence (AI)-enabled predictive maintenance framework for mobile cleaning robots to identify performance degradation and operational safety issues through vibration signals. A four-layer 1D CNN framework was developed and trained with a vibration signals dataset generated from the in-house developed autonomous steam mopping robot ‘Snail’ with different health conditions and hazardous operational environments. The vibration signals were collected using an IMU sensor and categorized into five classes: normal operational vibration, hazardous terrain induced vibration, collision-induced vibration, loose assembly induced vibration, and structure imbalanced vibration signals. The performance of the trained predictive maintenance framework was evaluated with various real-time field trials with statistical measurement metrics. The experiment results indicate that our proposed predictive maintenance framework has accurately predicted the performance degradation and operational safety issues by analyzing the vibration signal patterns raised from the cleaning robot on different test scenarios. Finally, a predictive maintenance map was generated by fusing the vibration signal class on the cartographer SLAM algorithm-generated 2D environment map.

Keywords: artificial intelligence; mobile cleaning robot; vibration source classification; predictive maintenance; deep learning; 1D CNN

1. Introduction

Mobile cleaning robots with various capacities are ubiquitous today, for instance in food courts, hypermarkets, hospitals, industries, airports, and homes, and are used for vacuuming, mopping, and sanitizing the environment. Market studies show that the personal and professional mobile cleaning robot growth is expected to reach 24 billion USD by 2026 [1]. However, proper maintenance and deployment in a robot-friendly workspace are crucial for autonomous mobile cleaning robots to avoid malfunction, catastrophic failure, and environmental-related safety issues, including customer dissatisfaction. Currently, manual supervision is widely used to monitor professional cleaning robots’ performance degradation and safety-related issues. However, it is time-consuming, labor and skill-set-dependent, and challenging to deploy due to the lack of historical failure data, especially for the newly developed advanced cleaning robots. Moreover, this periodical manual approach may trigger other issues such as extended downtime, the under-utilization of components, safety issues due to abrupt failure, and high operational and maintenance costs.

Automated predictive maintenance strategies overcome these pitfalls. They are widely used in industrial robots and autonomous vehicles for continuous health monitoring, performance degradation prediction, hazardous operational environment identification, and safety system failure indication. Various methods and techniques were proposed...
in the literature to implement automated predictive maintenance. A fuzzy inference approach is used in [2] to predict battery power status of robotics systems, and a non-intrusive methodology using torque sensor data for monitoring industrial robot joints in [3]. Similarly, a programmable motion-fault detection method for collaborative robots in [4], a framework to assess the future dynamic behavior and the remaining useful life of industrial robots in [5], and a data-driven predictive maintenance methodology using time-series electrical power data is used to detect manipulator errors in [6].

In recent years, artificial intelligence (AI) powered predictive maintenance (PdM) has been widely studied for automated PdM design. It adopts Machine Learning (ML) and Deep Learning (DL) algorithms for fault detection and classification. These works include a K-means clustering algorithm-based PdM for wafer transfer robot to avoid unplanned downtime proposed in [7], an automatic ML tool based health monitoring system to predict safe stops in a collaborative robot in [8], an Artificial Neural Network (ANN) model to predict the system failure of a packaging robot in [9], an ML-based PdM to detect drive belt looseness in a Cartesian robot in [10], and a DL-based fault diagnosis of industrial robots in [11] using multi-sensor fusion technology. Similarly, an ML-based fault diagnosis for the vehicle brake system is studied in [12] using wavelet applications, a terrain classification study for the autonomous ground vehicles is conducted in [13] adopting a probabilistic neural network, a hierarchical component-based diagnosis and prognosis system proposed for autonomous vehicles in [14] using a Dynamic Bayesian Network (DBN) model, and a DL-model developed to forecast the health of multi-sensor autonomous vehicles by training health index networks in [15].

Though several works are available for industrial robots and autonomous vehicle applications, predictive maintenance of autonomous mobile cleaning robots is not widely studied yet. The PdM system is a mandate function to autonomous mobile cleaning robots to deliver a safe and efficient service when operating in a complex and dynamic change environment, identify any performance degradation, and avoid operational safety issues. Generally, vibration is a key indicator for industrial robots and autonomous vehicles to predict performance degradation and hazardous operating environment identification, which is applicable for autonomous mobile cleaning robot platforms.

This work proposes an AI-powered predictive maintenance framework using vibration as a source for autonomous mobile cleaning robots. A one-dimensional Convolutional Neural Network (1D CNN) based vibration source classification model was developed and trained to classify the vibration signals arising from mobile cleaning robots. An in-house developed autonomous steam mopping robot—‘Snail’ is used to test and validate the proposed framework with different health conditions and hazardous operational environments.

The rest of this paper is organized as follows: Section 2 explains the literature survey on various vibration signal-based PdM works. Section 3 states the problem definition and motivation of this work. Section 4 gives an overview of the proposed system. Section 5 presents various experiments conducted and the results. Finally, Section 6 concludes the summary of the works.

2. Literature Survey

Currently, advanced industries use vibration-based health monitoring systems to detect early signs of the failure of machines and industrial robots. The vibration signals, measured using various vibration measuring sensors like piezoelectric or micro-electromechanical systems (MEMS) accelerometers, contain the health information of the equipment. In the literature, many reviews and research works have been conducted on vibration signal-based PdM using DL techniques, primarily for machine components, and industrial robots. The 1D CNN based vibration signal analysis studies for PdM includes a survey on 1D CNN benefits and applications in [16], fault diagnosis of machine components in [17–19], structural health monitoring in [20–22], and real-time fault diagnosis for power assets in [23]. Toh and Park reported the impact of vibration responses for early structural health monitoring in [24]. The study uses different DL architectures for vibration analysis. Pham
et al. applied CNN for fault diagnosis of bearings operated under different shaft speeds in [25]. They represented vibration signals as spectrograms to classify faults with high accuracy. Kolar et al. in [26] proposed a vibration signal-based fault diagnosis framework using the CNN algorithm. The authors used a three-axis accelerometer-generated vibration signal to detect and classify the faults. In [27], Chen et al. evaluated three deep neural networks models, including Deep Boltzmann Machines (DBM), Deep Belief Networks (DBN), and Stacked Auto-Encoders (SAE), to detect rolling bearing faults using vibration signals. Chen and Lee discussed in [28] a DL approach for vibration signal analysis for machining applications. Their study covers the optimisation method for CNN, 1D CNN, and 2D CNN structures with different types of inputs such as raw signal data and time-frequency spectra images. A DL model was developed by Luo et al. to detect early faults of a CNC machine in [29]. The authors used impulse responses from the vibration signals to detect the early mechanical fault under time-varying conditions. A fault diagnosis for the industrial robot was proposed by Wu et al. in [30]. Here, the authors combined three algorithms, including Manifold learning, Treelet Transform, and Naive Bayes, to detect the fault in industrial robots.

The studies mentioned above show that vibration signals are the decisive analyzing element, and DL-based techniques are suitable for the feature extraction from vibration signals for predicting a system’s performance degradation. However, most of the works available are focused on finding a specific fault, severity, or its remaining useful life, and do not consider the external factors for degradation. Moreover, the works are intended mainly for various machine components and industrial robots only. Hence, there is a research gap for monitoring autonomous cleaning robots’ health status and identifying hazardous environmental factors.

3. Problem Definition

A properly designed and developed cleaning robot works as required in its planned work environment without abnormal vibration. However, due to continuous operation or the impact of various internal and external environmental factors, the robot performance degrades and generates abnormal vibration signals. In most indoor cleaning robots, external terrain factors such as rough pebble pathways or tactile pavers produce high amplitude vibrations that cause performance degradation issues, such as assembly looseness, sensor misalignment, and faster component deterioration, for example. Vibration due to collisions with walls and other obstacles is an indicator of hazardous operation. It may arise due to the failure or malfunction of obstacle avoidance sensors or the misalignment of safety sensors, or the absence of hazardous object registration in the robot navigation map, such as tiny objects and glass walls (LiDAR sensor is sometimes not accurate in detecting tiny objects below object detection range or glass walls in [31–33]). A structural imbalance vibration signal is another indicator for both performance degradation and a hazardous operational environment. It may arise due to wheel damage, wear, a loose assembly of heavier components (battery, water tank), or a robot operating in a poor ground clearance area. Hence, identifying the vibration signal source is mandated to predict the cause of failure or performance degradation and identify the operational safety issues. It will enhance the predictive maintenance actions, for instance, isolating the potential hazardous region quickly, determining the severity and fixing the issues, redirecting to its intended workspace, securing a robot-friendly environment, and includes design improvement planning.

4. Overview of the Proposed System

Figure 1 shows an overview of the AI-enabled PdM framework for an autonomous cleaning robots platform. It uses vibration signals to predict cleaning robot performance degradation and any hazardous operational environment. The overview of the framework is explained as follows. This involves the robot platform details used for tests, the vibration data acquisition unit, the four-layer 1D CNN algorithm, the vibration source mapping module, and the remote monitoring unit.
4.1. Autonomous Steam Mopping Robot ‘Snail’

An in-house autonomous steam mopping robot (Snail) designed for cleaning and disinfection of indoor floors is used in this case study, as shown in Figure 2. The overall size of the robot is $40 \times 40 \times 38$ cm, and the total weight is 18 Kg, including inverter, battery, steam boiler, and mop head assembly. An NVIDIA Jetson AGX single-board computer is used to steer the entire operation of the robot, including autonomous navigation, executing predictive maintenance framework, and controlling all other sensors. In addition, the RPLIDAR A2 scanner is used for localization and mapping of the environment, and the Inertial Measurement Unit (IMU) Vectornav VN-100 sensor is used to estimate the motion, orientation, and heading angles of the robot. The robot locomotion was accomplished by a differential wheel drive mechanism with two supporting caster wheels. A D-Link 4G/LTE mobile router is used to control the robot remotely and monitor the robot’s health.

4.2. Vibration Data Acquisition Unit

In our study, five vibration classes are used as critical indicators for mobile cleaning robots’ performance degradation and operational safety issues. It is classified under three categories: normal, external factors, and internal factors, as shown in Figure 3. Here, terrain and collision-induced vibration belong to external factors, while assembly and structure-induced vibrations are due to internal factors.

Figure 1. Overview of the proposed DL-based PdM framework.

Figure 2. Autonomous steam mopping robot ‘Snail’.

Figure 3. Vibration source classification—Normal and Potential source of failure.
The linear and angular motion of the robot will be affected due to internal or external causes of vibration. Hence, we measured the linear acceleration (X, Y, and Z-axis) and angular velocity (roll, pitch, and yaw) of the robot using the onboard IMU sensor (VectorNav VN-100), which reflects the vibration level of the robot. Our IMU data subscription includes angular acceleration calculated from each instance’s current and previous angular velocity. Hence, the robot measures three signal data (linear acceleration, angular velocity, and angular acceleration) in three axes. This total of nine-sensor data collected during the exposure of five vibration source classes was used as the vibration signal data (feature values). The IMU subscription rate was set at 40 Hz, and each sample is grouped into 128 (time steps) data elements, which is every 3.2 s. The collected data is converted into a three-dimensional input array comprising samples, time steps, and features values. Figure 4 shows the data acquisition unit details, mainly the sensor position on the chassis at the center of the differential wheel drive axis.

![Figure 4. Data acquisition system and Linear-rotational motion of the Snail robot.](image)

4.3. 1D Convolutional Neural Network (1D CNN)

A 1D CNN model is adopted to build the proposed vibration source classification framework. The 1D CNN is formulated by convolution operations on data vectors as explained in [23]. It consists of signal input data vector $x$ (length $N$), a filter vector $\omega$ (length $L$), a bias term $b$ to best fit for given data, and a nonlinear activation function. The input vector is convolved with the filter vector, and its output layer is represented as in Equation (1). This representation of output layer $c$, of length $(N - L + 1)$ is without zero padding. To reduce the number of parameters and highlight the key feature, a max pooling output vector $d$ is defined, after each convolution layer, with a kernel size $m \times 1$, window function $u$, and filter moving stride $s$ as in Equation (2).

$$c(j) = f \left( \sum_{i=0}^{L-1} \omega(i)x(j-i) + b \right), j = 0, 1, \ldots, N - 1$$  (1)

$$d = \max(u(m \times 1,s)c)$$  (2)

Figure 5 shows the structure of the predictive maintenance 1D CNN framework. It involves mainly an input layer, four convolutional layers, a Fully Connected (FC) layer, and an output layer. The raw sensor data of the 3D array $[n \times 128 \times 9]$ gets normalised first. Then each sample is flattened to a 1D array $[1 \times 1152]$ for feeding into the CNN. The first two CNN layers use 64 filters and follow kernel size (convolution window) 3. The following two CNN layers use 32 filters and apply the same kernel size. A Rectified Linear Activation Unit (ReLU) is applied to each convolutional layer to learn complex nonlinear patterns in the signal data. After each convolutional layer, max pooling with a stride size of 2 and a dropout layer (dropout rate 0.2) is applied to reduce the feature map dimension and avoid over-fitting. Finally, a flattening function is used to convert the pooled feature map into a 1D array and pass it into the FC layer. A softmax layer is added as the final activation function in the output layer that predicts the multinomial probability. The output
The layer contains five neurons for the five vibration classes. Table 1 shows the details of the 1D CNN architecture.

| Layer            | Kernel Size | Stride | Filters | Data Shape |
|------------------|-------------|--------|---------|------------|
| Input            |             |        |         | (1152, 1)  |
| Conv. 1D-1       | 3 x 1       | 1      | 64      | (1150, 64) |
| Max Pool 1D-1    | 2 x 1       | 2 x 1  | 64      | (575, 64)  |
| Conv. 1D-2       | 3 x 1       | 1      | 64      | (573, 64)  |
| Max Pool 1D-2    | 2 x 1       | 2 x 1  | 32      | (287, 64)  |
| Conv. 1D-3       | 3 x 1       | 1      | 32      | (285, 32)  |
| Max Pool 1D-3    | 2 x 1       | 2 x 1  | 32      | (143, 32)  |
| Conv. 1D-4       | 3 x 1       | 1      | 32      | (141, 32)  |
| Max Pool 1D-4    | 2 x 1       | 2 x 1  |         | (71, 32)   |
| Fully Connected  |             |        |         | (100)      |
| Output (Softmax) |             |        |         | (5)        |

Table 1. 1D CNN Architecture.

4.4. Vibration Source Mapping Module

The mapping module performs 2D mapping of the environment using the onboard RPLidar and fuses the CNN predicted vibration source class on the 2D map generated. The Cartographer SLAM (Simultaneous Localisation and Mapping) algorithm is used to generate the 2D environment. It builds a grid-based map for the given environment. The prediction algorithm-generated vibration source classes are fused continuously into this grid map to generate a predictive maintenance map (PdM map). The user or maintenance team can visualize the type of performance degradation and safety-related issues on deployment space through the PdM map.

4.5. Remote Monitoring Unit

A smartphone app is developed to visualize the Snail robot’s real-time prediction results for remote health monitoring and control the robot in teleoperation mode, as shown in the overview layout Figure 1. The app is connected through the robot using the MQTT messaging protocol and collects the predicted information in request-based or continuous mode.

5. Experiments and Results

This section describes the experimental methods and results. The experiments were performed in four phases: dataset preparation, training the predictive maintenance CNN framework, validating the trained model with test dataset, and real-time field trials.
5.1. Data-Set Preparation and Pre-Processing

The dataset preparation involves collecting the vibration signals from the robot with different health states deployed on varying surface conditions, operational speeds, and cleaning patterns. Figure 6 shows the robot test set up for collecting the five classes of vibration including normal operational vibration, rough terrain induced vibration, collision-induced vibration, loose assembly induced vibration, and structure imbalance vibration.

Here, the normal operational and rough surface-induced vibrations were collected by deploying the robot over smooth indoor floors and rough terrain, respectively. Collision-induced vibrations were collected by hitting the robot on different obstacles, including walls and other static and dynamic (human) objects in the environment. The loose assembly-induced vibrations were collected by loosening the robot’s components, such as wheel coupling and mounting brackets. Finally, the unbalanced load-induced (structure) vibration data has been collected by using damaged/worn out wheels, asymmetrically placing heavier components, and operating the robot in a poor ground clearance area. The above mentioned five vibration classes were collected under different surface conditions (tile, concrete, carpet, wooden, vinyl, small and medium-size pebble pathways, and tactile pavers), operational speeds (linear 0.02–0.4 m/s and angular 0.3–1.3 rad/s), and cleaning patterns (straight, zig-zag, and spiral).

The Figures 7–11 shows the time-amplitude graph for the vibration signals raw data collected for all the five classes across each signal type (linear acceleration, angular velocity, angular acceleration) and its three-axis. The graphs provide a visual representation of how the signals vary through different vibration source classes. It is plotted for one sample (128 data), i.e., captured in 3.2 s.

Figure 6. Robot test set up for vibration data collection of five classes.

The Figures 7–11 shows the time-amplitude graph for the vibration signals raw data collected for all the five classes across each signal type (linear acceleration, angular velocity, angular acceleration) and its three-axis. The graphs provide a visual representation of how the signals vary through different vibration source classes. It is plotted for one sample (128 data), i.e., captured in 3.2 s.

Figure 7. Vibration signals—Normal class.
Data normalisation is applied in the pre-processing stage. The normalisation process involves bringing the raw data into a standard scale without losing information. In our case, the collected data \( x \) of all the nine features of each class is normalised into \(-1\) to \(+1\) using the Equation (3). Then, the pre-processed dataset is split into training, validation, and test data sets. The training and validation datasets are used to train the model, and the test dataset is used to evaluate the model after training. A total of 2500 samples for each class were recorded and split into 80% for training and 20% for validation. Furthermore, for evaluating the model, a total of 500 samples were collected for each class as test data sets.

\[
    x_{\text{Normalised}} = 2 \frac{x - \min(x)}{\max(x) - \min(x)} - 1 \tag{3}
\]

5.2. Training and Validation

A supervised learning strategy was used in this PdM framework using our unique dataset. Tensorflow [34] deep learning library was used to develop the predictive maintenance CNN framework, and Nvidia GeForce GTX 1080 Ti-powered workstation was used to train the model with the collected dataset. Table 2 shows the hyperparameter settings for training the model. Momentum with gradient descent was used as the optimising strategy to speed up learning and not get stuck with local minima. Adam optimiser (adaptive moment optimization) [35] with three parameters showed better training results: a learning rate of 0.001, the exponential decay rate for the first moment of 0.9, and for the second moment of 0.999. Different epochs were used for testing, and better accuracy was found with 100 epochs and a batch size of 32. The categorical cross-entropy loss function is used while compiling the model to reduce the loss during training and to improve prediction probability.
In the training phase, a K-fold (in our study, K = 5) cross-validation technique is used to evaluate the dataset’s quality, improve generalization, avoid over-learning, and choose the best model for this application. In k-fold cross-validation, the datasets are split into k subsets and k-1 subsets to train the model. The remaining one is for evaluating the model’s performance.

5.3. Prediction with Test Dataset

The vibration class prediction efficiency of the trained model is evaluated with the test dataset. A total of 500 test samples were used for each class to evaluate the model. These test datasets have not been used in the training and cross-validation of the model. Accuracy, precision, recall, and F1 Score (Equations (4)–(7)) statistical measure metrics [36] were used to assess the model performance. Here, $TP, FP, TN, FN$ represents the true positives, false positives, true negatives, and false negatives, respectively, as per the standard confusion matrix.

\[
\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{4}
\]

\[
\text{Precision} = \frac{TP}{TP + FP} \tag{5}
\]

\[
\text{Recall} = \frac{TP}{TP + FN} \tag{6}
\]

\[
F1\text{Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{7}
\]

Table 3 shows the statistical measure result for the test dataset. Accordingly, the model classifies the normal operational vibration class with 89% accuracy, and the hazardous terrain-induced vibration and collision-induced vibration accuracy were predicted with 95% and 92% accuracy, respectively. Furthermore, loose assembly-induced vibration and structure imbalance vibration classes were predicted with 94% and 91% accuracy. The model’s overall classification accuracy (average of five classes) is 92.2%. The above analysis shows that the trained model has accurately classified the five vibration classes collected from the mobile cleaning robot run on different surfaces. Hence, this proposed model is suitable for real-time deployment in mobile cleaning robots for performance degradation and hazardous operational region prediction.

5.4. Comparison with Other Algorithms

The performance of the proposed predictive maintenance 1D CNN framework was compared with other commonly used ML/DL classifier models, such as Support Vector Ma-

### Table 2. HyperParameters setting.

| Parameter       | Values/Function |
|-----------------|-----------------|
| Optimizer       | Adam            |
| Learning rate   | 0.001           |
| Batch Size      | 32              |
| Epochs          | 100             |

### Table 3. Offline test result.

| Vibration Source | Precision | Recall | F1 Score | Accuracy |
|------------------|-----------|--------|----------|----------|
| Normal           | 0.86      | 0.90   | 0.89     | 0.89     |
| Terrain          | 0.97      | 0.95   | 0.96     | 0.95     |
| Collision        | 0.92      | 0.92   | 0.92     | 0.92     |
| Assembly         | 0.93      | 0.94   | 0.94     | 0.94     |
| Structure        | 0.86      | 0.92   | 0.93     | 0.91     |
chine (SVM) [37], Multilayer Perceptron (MLP) [38], Long Short-Term Memory (LSTM) [39], and CNN-LSTM [40]. The same training and test dataset, processing resources, and conditions used for the 1D CNN model have been applied for a fair comparison. The CNN-LSTM, LSTM, and MLP models were trained using the TensorFlow library and SVM with Scikit-learn [41] package. The key parameter settings such as optimiser (Adam), learning rate (0.001), and loss function (categorical cross-entropy) for the CNN-LSTM, LSTM, and MLP comparison models were used the same as 1D CNN. For the SVM comparison model, the key parameters ‘C’ and ‘gamma’ values used 100 and 0.01, respectively, and the Radial Basis Function (RBF) kernel was applied. The overall accuracy of each model over five classes and the inference time (millisecond) to process one sample data are given in Table 4.

Table 4. Accuracy comparison with other models.

| Model      | Accuracy (%) | Inference Time (ms) |
|------------|--------------|---------------------|
| 1D CNN     | 92.2         | 0.162               |
| CNN-LSTM   | 88.1         | 0.258               |
| LSTM       | 85.4         | 0.276               |
| MLP        | 79.8         | 0.193               |
| SVM        | 77.5         | 1.675               |

The comparison results show that our proposed predictive maintenance 1D CNN framework has scored better classification accuracy and took less inference time than the other four algorithms. Hence, it is evident that our proposed system is an optimal algorithm to predict the performance degradation and hazardous operational environment.

5.5. Real-Time Prediction

The real-time prediction experiment was tested with the Snail robot in four different environments at the SUTD campus include the lobby, food court, corridor, and lab workspace. These environments were not used for the training data-set collection process. Before the real-time prediction experiment, all the environments were mapped by the cartographer SLAM algorithm using the on-board RPLIDAR A2 for autonomous navigation and predictive maintenance map generation. In our experiment, the continuous field trial was conducted on multiple days to observe the robot performance degradation and hazardous operational prediction. The trained model was configured in robot onboard computer NVIDIA Jetson AGX. Its prediction results (terrain, collision, assembly, or structure-induced vibrations signals) were fused into the cartographer SLAM generated environment map with different colors to identify performance degradation and hazardous operational prediction.

The first case study was trialed in a lobby environment consisting of a glass sidewall and carpet floor. Here, collision-induced vibrations were registered in the cartographer SLAM algorithm-generated map due to incorrect mapping of the glass wall. The glass wall was covered with a raising curtain in the original map during the mapping time, whereas it was removed while testing. As a result, the robot lost the previously mapped navigation awareness, and RPLidar could not locate the glass as an obstacle. The robot hit on the glass randomly, and the repeated collision-induced vibration marks were captured in the lobby environment 2D map as shown in Figure 12a.
The SUTD food court was our second case study testbed. The food court environment has smooth concrete flooring with dining furniture. The robot trial was conducted during different operational times, including peak hours. Figure 12b shows the SUTD food court set up and its cartographer SLAM map registered with predicted vibration signals. Here, the collision-induced vibration and hazardous surface-induced (terrain) vibrations were recorded in the environment map. Specifically, collision-induced vibration was registered primarily during peak business hours, which arises due to accidental collision with humans, undetected furniture legs, and changes in the position of the dining furniture. Similarly, tactile pavers and cables set on the floor caused hazardous terrain-induced vibrations.

Case study three was conducted in a corridor environment with mixed style flooring with a smooth concrete floor, and a pebble paved rough surface. During our experiment, the robot was first deployed to clean the smooth concrete floor. Here, no abnormal vibrations were reported in the 2D environment map. Later, when the robot started cleaning rough pebbled surfaces, hazardous terrain-induced vibrations marks were seen on the map. When the robot was deployed for a ten day trial on the pebbled surface, a mix of terrain and loose assembly-induced vibrations were registered, as shown in Figure 12c. This is mainly due to the loose assembly of mechanical systems such as loosening of mounting brackets screws and deterioration of mop cloth.

The fourth case study was performed in our SUTD ROAR lab workspace, where the environment filled dynamic objects and the terrain consisted of vinyl flooring. The experiment lasted for four weeks, and its cumulative results in the lab environment were observed. In the first three weeks, the map showed regular drive without capturing any abnormal vibration signature. However, the fourth-week map showed a loose assembly pattern observed from the wobbled wheels due to the loosening of the wheel coupling set screws. The test continued in this wheel-wobbled state and noticeable imbalanced structural vibrations arose due to the battery becoming detached from its bracket and creating an unbalanced weight distribution. Figure 12d shows the four-week test results for case study four, where the environment map depicts loose assembly and unbalanced weight structural patterns.

As per the experimental results, we observed that one source of failure might lead to another if no action is taken. This way, if several sources of failure are present, the model will predict the predominant failure of the robot in that particular instance. Hence, through the mobile app, the maintenance person can remotely stop the robot as soon as the
initial abnormal class is registered, avoiding chances for multiple failures or hazards to the environment.

Table 5 shows the statistical measure result for 100 test samples collected from the real-time case study. Here, the sample data were collected from the Snail robot through the mobile app on request-based sample mode and performed the statistical measure using the confusion matrix function. The table results indicate that the algorithm classified the five different vibration sources with an average accuracy of 88.9%, 93.5%, 91.8%, 92.1%, and 88.7% for normal, terrain, collision, assembly, and structure, respectively. In contrast with offline results, the model average prediction accuracy of five classes in real-time tests is slightly less (91%) due to external noise and various sampling periods. However, it can be reduced by fusing multiple IMU sensors or adding a noise cancellation function in the preprocessing stage.

Table 5. Real-time prediction accuracy of five classes.

| Vibration Source | Normal | Terrain | Collision | Assembly | Structure |
|------------------|--------|---------|-----------|----------|-----------|
| Prediction (%)   | 88.9   | 93.5    | 91.8      | 92.1     | 88.7      |

6. Conclusions

An AI-enabled predictive maintenance framework was proposed for mobile cleaning robots to monitor performance degradation and identify operational safety issues. The proposed framework was tested and validated with our in-house developed autonomous steam mopping robot ‘Snail’. A four-layer 1D CNN model was developed using TensorFlow API and trained with five vibration signal datasets generated from the Snail robot with different health conditions. The efficiency of the proposed predictive maintenance framework was evaluated with offline and real-time field tests. The experimental results show that the model scored 92.2% accuracy for classifying the performance degraded and hazardous operational vibration signals in offline tests and took 0.162 ms to process one test sample. In the real-time field test, the algorithm accurately predicted robot performance degradation and operational safety issues with an accuracy of 91%. The predicted vibration signal class was fused into the cartographer SLAM-generated environment map to track the performance degradation and identify the operational safety issues. This will help manufacturers and cleaning maintenance companies to choose the right maintenance strategy, rental policy, or improve the robot design and assembly.

Author Contributions: Conceptualization, S.P.; Methodology, S.P. and B.R.; Software, V.S. and S.P.; Validation, S.P.; Formal analysis, B.R. and S.P.; Investigation, B.R. and S.P.; Resources, M.R.E.; Data curation, S.P. and V.S.; Writing—original draft, S.P. and B.R.; Writing—review & editing, B.R. and S.P.; Visualization, S.P.; Supervision, M.R.E.; Project administration, M.R.E.; Funding acquisition, M.R.E. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the National Robotics Programme under its Robotics Enabling Capabilities and Technologies (Funding Agency Project No. 192 25 00051), National Robotics Programme under its Robotics domain Specific (Funding Agency Project No. 192 22 00058, 192 22 00108) and administered by the Agency for Science, Technology and Research.

Acknowledgments: The authors would like to thank, the National Robotics Programme, the Agency for Science, Technology and Research and SUTD for their support.

Conflicts of Interest: There are no conflict of interest.
References

1. Research and Markets. Worldwide Cleaning Robot Industry to 2026-Key Market Drivers and Restraints. Available online: https://www.prnewswire.com/news-releases/worldwide-cleaning-robot-industry-to-2026---key-market-drivers-and-restraints-301293632.html (accessed on 10 June 2021).

2. Huang, H.P.; Wu, S.H. Diagnostic and predictive maintenance systems for abnormal behavior of power scheduling loading and its application to robotics systems. In Proceedings of the 2011 9th World Congress on Intelligent Control and Automation, Taipei, Taiwan, 21–25 June 2011; pp. 908–913. [CrossRef]

3. Izgirre, U.; Andonegui, I.; Egea, A.; Zurutuza, U. A methodology and experimental implementation for industrial robot health assessment via torque signature analysis. Appl. Sci. 2020, 10, 7883. [CrossRef]

4. Park, Y.S.; Yoo, D.Y.; Lee, J.W. Programmable Motion-Fault Detection for a Collaborative Robot. IEEE Access 2021, 9, 133123–133142. [CrossRef]

5. Aivaliotis, P.; Arkouli, Z.; Georgoulias, K.; Makris, S. Degradation curves integration in physics-based models: Towards the predictive maintenance of industrial robots. Robot. Comput.-Integr. Manuf. 2021, 71, 102177. [CrossRef]

6. Borgi, T.; Hidri, A.; Neef, B.; Naceur, M.S. Data analytics for predictive maintenance of industrial robots. In Proceedings of the 2017 International Conference on Advanced Systems and Electric Technologies (IC_ASET), Hammamet, Tunisia, 14–17 January 2017; pp. 412–417.

7. Kim, H.G.; Yoon, H.S.; Yoon, J.H.; Yoon, H.I.; Han, S.S. Development of Predictive Maintenance Technology for Wafer Transfer Robot using Clustering Algorithm. In Proceedings of the 2019 International Conference on Electronics, Information, and Communication (ICEIC), Auckland, New Zealand, 22–25 January 2019; pp. 1–4.

8. Aliev, K.; Antonelli, D. Proposal of a Monitoring System for Collaborative Robots to Predict Outages and to Assess Reliability Factors Exploiting Machine Learning. Appl. Sci. 2021, 11, 1621. [CrossRef]

9. Onur, K.; Kaymakci, O.T.; Mercimek, M. Advanced Predictive Maintenance with Machine Learning Failure Estimation in Industrial Packaging Robots. In Proceedings of the 2020 International Conference on Development and Application Systems (DAS), Suceava, Romania, 21–23 May 2020; pp. 1–6.

10. Pierleoni, P.; Belli, A.; Palma, L.; Sabbatini, L. Diagnosis and Prognosis of a Cartesian Robot’s Drive Belt Looseness. In Proceedings of the 2020 IEEE International Conference on Internet of Things and Intelligence System (IoTaIS), BALI, Indonesia, 27–28 January 2021; pp. 172–176.

11. Wang, J.; Wang, D.; Wang, X. Fault diagnosis of industrial robots based on multi-sensor information fusion and 1D convolutional neural network. In Proceedings of the 2020 39th Chinese Control Conference (CCC), Shenyang, China, 27–29 July 2020; pp. 3087–3091.

12. Alamelu Manghai, T.; Jegadeeshwaran, R. Vibration based brake health monitoring using wavelet features: A machine learning approach. J. Vib. Control 2019, 25, 2534–2550. [CrossRef]

13. Dupont, E.M.; Moore, C.A.; Collins, E.G.; Coyle, E. Frequency response method for terrain classification in autonomous ground vehicles. Auton. Robot. 2008, 24, 337–347. [CrossRef]

14. Gomes, I.P.; Wolf, D.F. Health Monitoring System for Autonomous Vehicles using Dynamic Bayesian Networks for Diagnosis and Prognosis. J. Intell. Robot. Syst. 2021, 101, 1–21. [CrossRef]

15. Safari, S.; Safari, M.A.; Hamid, H.; Fallah, S. Multi-Sensor Fault Detection, Identification, Isolation and Health Forecasting for Autonomous Vehicles. Sensors 2021, 21, 2547. [CrossRef]

16. Kiranyaz, S.; Avci, O.; Abdeljaber, O.; Ince, T.; Gabbouj, M.; Inman, D.J. 1D convolutional neural networks and applications: A survey. Mech. Syst. Signal Process. 2021, 151, 107398. [CrossRef]

17. Eren, L.; Ince, T.; Kiranyaz, S. A generic intelligent bearing fault diagnosis system using compact adaptive 1D CNN classifier. J. Signal Process. Syst. 2019, 91, 179–189. [CrossRef]

18. Abdeljaber, O.; Sassi, S.; Avci, O.; Kiranyaz, S.; Ibrahim, A.A.; Gabbouj, M. Fault detection and severity identification of ball bearings by online condition monitoring. IEEE Trans. Ind. Electron. 2018, 66, 8136–8147. [CrossRef]

19. Ince, T.; Kiranyaz, S.; Eren, L.; Askar, M.; Gabbouj, M. Real-time motor fault detection by 1-D convolutional neural networks. IEEE Trans. Ind. Electron. 2016, 63, 7067–7075. [CrossRef]

20. 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. [CrossRef]

21. Abdeljaber, O.; Avci, O.; Kiranyaz, M.S.; Boashash, B.; Sodano, H.; Inman, D.J. 1-D CNNs for structural damage detection: Verification on a structural health monitoring benchmark data. Neurocomputing 2018, 275, 1308–1317. [CrossRef]

22. Avci, O.; Abdeljaber, O.; Kiranyaz, S.; Hussein, M.; Inman, D.J. Wireless and real-time structural damage detection: A novel decentralized method for wireless sensor networks. J. Sound Vib. 2018, 424, 158–172. [CrossRef]

23. Mitiche, I.; Nesbitt, A.; Conner, S.; Boreham, P.; Morison, G. 1D-CNN based real-time fault detection system for power asset diagnostics. IET Gener. Transm. Distrib. 2020, 14, 5766–5773. [CrossRef]

24. Toh, G.; Park, J. Review of vibration-based structural health monitoring using deep learning. Appl. Sci. 2020, 10, 1680. [CrossRef]

25. Pham, M.T.; Kim, J.M.; Kim, C.H. Accurate bearing fault diagnosis under variable shaft speed using convolutional neural networks and vibration spectrogram. Appl. Sci. 2020, 10, 6385. [CrossRef]

26. Kolar, D.; Lisjak, D.; Pajak, M.; Pavković, D. Fault diagnosis of rotary machines using deep convolutional neural network with wide three axis vibration signal input. Sensors 2020, 20, 4017. [CrossRef]
27. Chen, Z.; Deng, S.; Chen, X.; Li, C.; Sanchez, R.V.; Qin, H. Deep neural networks-based rolling bearing fault diagnosis. *Microelectron. Reliab.* 2017, 75, 327–333. [CrossRef]

28. Chen, H.Y.; Lee, C.H. Deep Learning Approach for Vibration Signals Applications. *Sensors* 2021, 21, 3929. [CrossRef] [PubMed]

29. Luo, B.; Wang, H.; Liu, H.; Li, B.; Peng, F. Early fault detection of machine tools based on deep learning and dynamic identification. *IEEE Trans. Ind. Electron.* 2018, 66, 509–518. [CrossRef]

30. Wu, Y.; Fu, Z.; Fei, J. Fault diagnosis for industrial robots based on a combined approach of manifold learning, treelet transform and Naive Bayes. *Rev. Sci. Instrum.* 2020, 91, 015116. [CrossRef]

31. Tibeju, H.; Roche, J.; De Silva, V.; Kondoz, A. LiDAR-Based Glass Detection for Improved Occupancy Grid Mapping. *Sensors* 2021, 21, 2263. [CrossRef] [PubMed]

32. Foster, P.; Sun, Z.; Park, J.J.; Kuipers, B. Visagge: Visible angle grid for glass environments. In Proceedings of the 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 6–10 May 2013; pp. 2213–2220.

33. Schwartz, M.; Zarzycki, A. The effect of building materials on LIDAR measurements. In Proceedings of the 35th eCAADe Conference-Volume 2, Rome, Italy, 20–22 September 2017.

34. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; et al. Tensorflow: A system for large-scale machine learning. In Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), Savannah, GA, USA, 2–4 November 2016; pp. 265–283.

35. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* 2014, arXiv:1412.6980.

36. Grandini, M.; Bagli, E.; Visani, G. Metrics for multi-class classification: An overview. *arXiv* 2020, arXiv:2008.05756.

37. Marius, H. Multiclass Classification with Support Vector Machines (SVM) Dual Problem and Kernel Function. *Towards Data Sci.* 2020. Available online: https://towardsdatascience.com/multiclass-classification-with-support-vector-machines-svm-kernel-trick-kernel-functions-f9d5377d6f02 (accessed on 8 October 2021).

38. Taud, H.; Mas, J. Multilayer perceptron (MLP). In Geomatic Approaches for Modeling Land Change Scenarios; Springer: Berlin/Heidelberg, Germany, 2018; pp. 451–455.

39. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* 1997, 9, 1735–1780. [CrossRef]

40. Mohamad, T.H.; Abbasi, A.; Kim, E.; Nataraj, C. Application of Deep CNN-LSTM Network to Gear Fault Diagnostics. In Proceedings of the 2021 IEEE International Conference on Prognostics and Health Management (ICPHM), Detroit (Romulus), MI, USA, 7–9 June 2021; pp. 1–6.

41. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* 2011, 12, 2825–2830.