Hindawi
IET Signal Processing
Volume 2023, Article ID 4721508, 13 pages
https://doi.org/10.1049/2023/4721508

Research Article

Robot Ground Media Classification Based on Hilbert–Huang Transform and Attention-Based Spatiotemporal Coupled Network

Jixiang Niu,[1] **Han Li**,[1] **Zhenxiong Liu**,[1] **Wei Liu**,[1] **and Hejun Xu**[2,3]

[1] *North China University of Technology, Beijing 100144, China*
[2] *Department of Architecture and Civil Engineering, City University of Hong Kong, Hong Kong 999077, China*
[3] *School of Civil Engineering and Architecture, Jiangsu University of Science and Technology, Zhenjiang, Jiangsu 212003, China*

Correspondence should be addressed to Hejun Xu; hejunxu2-c@my.cityu.edu.hk

Received 6 May 2023; Revised 28 June 2023; Accepted 26 July 2023; Published 23 October 2023

Academic Editor: Wenling Li

With the development of technology, mobile robots are increasingly deployed in real-world environments. To enable robots to work safely in a variety of terrain environments, we proposed a ground-type detection method based on the Hilbert–Huang transform (HHT) and attention-based spatiotemporal coupled network. Taking a dataset containing multiple sets of robot signals from a Kaggle competition as an example; we use the proposed method to classify the signals and thus achieve a terrain classification of the robot's location. Firstly, the signal data were processed using the discrete wavelet transform for noise reduction, and all channels in the dataset were ranked by importance using the permutation importance method. Next, the instantaneous frequencies of the two most important channels were extracted using the HHT and added to the original dataset to expand the feature dimension. Then the features in the expanded dataset were extracted by the convolutional neural network, long short-term memory, and attention module. Afterward, the fully extracted features were passed into the fully connected layer for classification, and an average classification accuracy of 83.14% was obtained. The effectiveness of each part in our method was demonstrated using ablation experiments. Finally, we compared our method with some common methods in the field and found that our method obtained the highest classification accuracy, proving the superiority of the proposed method.

1. Introduction

With the development of artificial intelligence technology, mobile robots are increasingly deployed in various domains, including forestry, mining, rescue, and space exploration [1]. Robots may encounter complex, unknown, and even dangerous terrain that may cause it to be damaged or down. For example, if a robot encounters slippery terrain while traveling at high speeds, accidents such as skidding and sinking may occur [2, 3], or if it encounters bumpy sections and rocks, damage can be caused to the robot [4]. On the other hand, detailed information about the ground medium allows robots to work more effectively in the real world [5]. Therefore, there is an urgent need for a way to detect the ground in the environment where the mobile robot is located, allowing the robot to adapt its driving style and strategies to different terrains to accomplish its tasks more efficiently and safely.

Many scholars have explored applying terrain perception in mobile robots. The mainstream methods can be broadly divided into two categories. The first methods category is more traditional and primarily includes using cameras or radar to intuitively obtain relevant information about the terrain near the robot and calculate terrain parameters using relevant mechanical knowledge. These methods are simple to apply but are subject to more significant external interference and have relatively low terrain classification accuracy [6–8]. The second category of methods is primarily based on sensors collecting signals of robot–terrain interactions. The collected signals are analyzed in time domains to extract relevant features, significantly improving the classification effect [9–12].

Most of the early-stage studies have used the first type of approach mentioned above. Howard and Seraji [6] developed a set of vision algorithms that extracted features from image data obtained from cameras mounted on robots and combined the features to form a fuzzy traversability metric that quantified the ease of mobile robots traversing over terrain. However, the accuracy of vision-based algorithms is significantly reduced when visibility is relatively low.

Lalonde et al. [7] used radar to classify terrain. This approach emphasizes terrain segmentation based on obstacles rather than the terrain itself and is still subject to weather interference. Iagnemma et al. [8] proposed an algorithm that relies on classical terrain mechanic equations and uses linear least squares to calculate terrain parameters in real time, which can successfully calculate terrain parameters but is still subject to noise interference. In short, it is a challenging task to sense terrain reliably.

Inspired by how humans and other animals perceive terrain through their footsteps [13], the primary approach in the current research stage has shifted from the previous traditional methods to terrain perception using acoustic and vibration responses generated by the robot–terrain interaction. At the same time, with the continuous improvement and development of sensor-related technologies and performance, sensors have been applied in many applications for signal collection [9]. Zhao et al. [10] proposed a new classification framework using acoustic and vibration signals generated by robot–terrain interaction and selected multiple data sets for complementary experiments, which achieved good classification results. Wu et al. [11] used a miniature capacitive tactile sensor array to collect ground pressure data by directly measuring the ground reaction force on the legs of a small running robot, which was used as input data for a support vector machine (SVM) classifier and subsequent ground classifications. However, the above method still suffers from two problems: (1) the collected signals are primarily affected by background noise, and (2) the signals have a mixed distribution between categories. These problems make it difficult to extract features directly from signals.

Some standard time–frequency analysis methods [14] can reduce noise and improve the signal-to-noise ratio. Standard analysis methods include the Fourier transform and wavelet transform. Further, The wavelet transform is divided into the continuous wavelet transform (CWT) and the discrete wavelet transform (DWT). In 1997, Hazarika et al. [15] proposed using the Wavelet Transform to preprocess EEG (electroencephalogram) signals before classifying them as normal or abnormal. Experiments showed that the network trained with wavelet coefficients could correctly classify EEGs in the normal and schizophrenia classes with 66% and 71% accuracy, respectively. Saravanan and Ramachandran [16] used the DWT to extract all the signals' wavelet coefficients and features, performed feature selection on various discrete wavelets and used the wavelets with maximum potential as input for a neural network to obtain a subsequent classification. Deokar and Waghmare [17] proposed a DWT-FFT (fast Fourier transform) method using integration rules to integrate the traditional DWT with an FFT for classification. The results showed that the classifier based on the DWT-FFT method could obtain high accuracy with less computational complexity.

However, both the Fourier transform and the wavelet transform have their own limitations. For the former, it is mainly applied to smooth signals, but the signals collected for this study may contain nonsmooth or transient features. For the latter, although the wavelet transform can theoretically handle nonlinear nonsmooth signals, it can only navigate linear nonsmooth signals in practical algorithmic implementations.

The Hilbert–Huang transform (HHT) [18] differs from these traditional methods in that it is entirely free from linearity and smoothness constraints and is suitable for analyzing nonlinear nonsmooth signals. Fu et al. [19] used the HHT for processed EEG signals and an SVM classifier to classify whether epilepsy presented with seizures. It was experimentally demonstrated that the method could achieve the best average classification accuracy. Guo et al. [20] transformed sampled signals into spectrograms using the HHT. They later used convolutional neural networks (CNNs) to extract image features to perform fault classification in power distribution systems successfully.

The systems used in previous works send all extracted features directly to classifiers. Still, the distributions of signal patterns collected by the sensors vary greatly, so there is usually a mixed distribution between categories. To solve this problem and improve the accuracy of terrain classification, signal patterns need to be passed into the classifier for further feature extraction and final ground classification. In previous works, the methods that appear most frequently can be broadly classified into two categories. The first category is based on traditional machine learning classification models, such as decision trees and SVMs [11, 12, 19, 21–23]. The second category is based on deep learning with various neural networks, such as CNNs and artificial neural networks (ANNs) [16, 20, 24].

The traditional SVM machine learning model can solve convex optimization problems with globally optimal solutions and has a low Vapnik-Chervonenkis dimensionality, which allows for classifying high-dimensional data with fewer optimization parameters, making the model popular among scholars. In 2013, Subasi [21] proposed a method that combines the particle swarm optimization (PSO) algorithm and SVM, which produced an overall classification accuracy of 97.41% for 1200 EMG signals selected from 27 topics. Du and Zhu [22] proposed a PCA (principal component analysis)-SVM method that included feature fusion and dimensionality reduction using PCA before passing data into an SVM classifier for ground classification. In addition to SVM, other machine learning models can be applied to related tasks. Gokgoz and Subasi [23] proposed an EMG signal classification framework using a multiscale PCA for noise reduction, the DWT for feature extraction, and a decision tree algorithm for classification. Comparing the effects of several different decision tree algorithms, the results showed that the combination of the DWT and random forest machine learning model obtains the best performance and the highest total classification accuracy.

With the increase in computing power, deep learning, and neural networks have been widely used for various tasks in recent years. Saravanan and Ramachandran [16] used an ANN for subsequent classification after extracting features with the DWT and noise reduction. They experimentally found that neural networks have a high potential in condition-monitoring gearboxes with various faults. Baishya and Bauml

[24] found that these methods were not robust after using K-nearest neighbor, SVM, and Bayesian analyses after a dimensionality reduction of the features. So they decided to use a CNN for high dimensional data and found that the CNN outperformed the previous best classifier. After transforming surface electromyography signals into images, Duan et al. [25] used a CNN to classify images. They then completed the classification of limb movements and found that a CNN showed better robustness than other methods. Prabhakar et al. [26] used partial least squares nonlinear regression technique, expectation–maximization-based PCA technique, and isometric mapping (Isomap) technique to extract features, which were later further optimized by four optimization algorithms (pollen algorithm, hawk strategy using different evolutionary algorithms, backtracking search optimization algorithm and group search optimization algorithm) to provide ideas for the treatment of related problems. Liu et al. [27] proposed a framework for signal classification based on deep learning networks, where signals are preprocessed and represented as 2D time–frequency images by Choi–Williams distributed time–frequency analysis. Corresponding simulation results show that the proposed framework is able to learn the hierarchical features accurately and achieve excellent signal classification performance.

The motion of a robot is a continuous process, and time is the basis of many intrinsic behaviors. Signals of a last moment and the next moment are necessarily related, and the ANN and CNN methods mentioned above can extract features in the spatial dimension. Still, it is difficult to mine backward and forward connections in time. Therefore, recurrent neural networks (RNNs) must be used to solve the problem in this paper. However, spatial relationships exist between the information collected by multiple sensors simultaneously in this task. Thus, it is difficult to combine this spatial information effectively using only RNNs. Therefore, a ground classification detection method based on the HHT and attention-based spatiotemporal coupled network is proposed in this study. After the DWT denoise, the instantaneous frequency features were extracted by the HHT to expand the dataset. Then the CNN, long short-term memory (LSTM), and attention module were sequentially used to achieve the final ground-type classification.

Specifically, the main contributions of this paper are as follows:

(1) We used DWT to preprocess the input signal for noise reduction and compared the effect of several different sets of denoise thresholds on the final results.

(2) To analyze the signals from various angles and extract the signal features more accurately, the HHT was used to process signals. Unlike the traditional idea of image classification using HHT spectrograms, we expand the features while preserving the original data. Compared with image classification, our method is less expensive, more accurate, and less likely to be overfitted.

(3) In our dataset, different groups of signals from the same sensor were collected at different moments. The signals collected at the same moment came from different sensors. Therefore, a combination of CNN and LSTM was used to realize the coupled calculation of temporal and spatial information.

(4) An attention mechanism was introduced after the LSTM layer to enhance the weight of important information, reduce the number of parameters, and improve efficiency during the training process.

This paper is organized in the following manner. Section 1 introduces the background of the problem and related work; Section 2 describes the rationale of each method used in this study. Section 3 explains the working process of each experiment and presents the results. Finally, Section 4 concludes the paper and provides possible future research directions.

2. Methods

In brief, our model can be divided into a denoise module, a feature extraction module, and a classification module. In the denoise module, the discrete wavelet transform is used to denoise the input data, which can make the signal features in the dataset more obvious and convenient for subsequent feature extraction. In the feature extraction module, the HHT is first used to extract the instantaneous frequencies of the two most important channels. Next, the dataset is enlarged with the extracted frequency information from the first step. Then, features from the expanded dataset are fully extracted by combing the spatial feature extraction capability of 1D CNN and the temporal feature capture capability of LSTM. Finally, an attention mechanism is introduced after the LSTM layer to enhance the extraction effect of the network. In the classification model, the extracted features are passed through a fully connected layer to achieve the final terrain classification.

2.1. Denoise by the Discrete Wavelet Transform.

The function $f(t)$ in any L2 (R) space is expanded under the wavelet basis, and this expansion is called the CWT of the function $f(t)$. Its expression is given by the following:

$$WT_f(a,\tau) = <f(t), \psi_{a,\tau}(t)> = \frac{1}{\sqrt{a}} \int_R f(t)\psi^*\left(\frac{t-\tau}{a}\right)dt. \quad (1)$$

For continuous wavelets, the scale parameter a, the time t, and the time-shift parameter τ are all continuous. When using the computer to calculate these three parameters, they must be discretized to obtain the discrete wavelet transform. Specifically, the discrete wavelet transform discretizes the scale parameter a and the time shift parameter τ of the continuous wavelet transform, but t is still continuous.

The principle of discrete wavelet denoising is that the wavelet coefficients of the signal are larger after wavelet decomposition, while the wavelet coefficients of the noise are smaller. Therefore, a suitable choice of threshold value

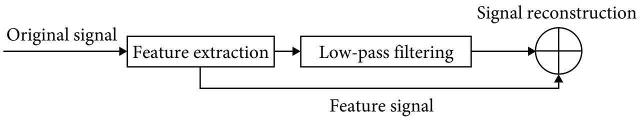

Figure 1: DWT denoising diagram.

is required to judge whether a wavelet coefficient is generated by the signal or the noise. That is, the wavelet coefficients larger than the threshold value are considered to be generated by the signal and need to be retained; those wavelet coefficients smaller than the threshold value are considered to be generated by the noise and are set to zero to achieve the purpose of denoising. This method can successfully retain the signal characteristics after denoising, so it is better than the traditional low-pass filter. The diagram of DWT is shown in Figure 1 and our model diagram is shown in Figure 2.

2.2. HHT. In signal science, the Hilbert transform can directly calculate the instantaneous amplitude, the instantaneous frequency, and the phase of the signal. However, the instantaneous frequency calculated by this method may contain negative values that do not have real physical meaning. The HHT transform first decomposes the signal using empirical mode decomposition (EMD) to obtain the components at different scales and then performs the Hilbert transform on each component to obtain the instantaneous frequency with real meaning, thus realizing the high-resolution time–frequency analysis. In the experiment, each data set is a physically meaningful timing signal collected by sensors. Therefore, the HHT transform is very suitable for our task.

2.2.1. EMD. EMD [18] is the process of decomposing a complex signal into a finite number of intrinsic mode functions (IMFs) based on the time-scale characteristics of the data itself, without any predetermined basis functions. The decomposed IMF components contain the local feature information of the original signal at different time scales, which is a kind of time–frequency domain signal processing method. Specifically, EMD decomposes the input signal into several intrinsic mode functions and one residual part, i.e., for each input signal, there are as follows:

$$I(n) = \sum_{m=1}^{M} IMF_m(n) + \mathrm{Re}\, s_M(n), \tag{2}$$

where $IMF_m(n)$ denotes the mth eigenmode function and $\mathrm{Res}_M(n)$ denotes the residual. For each of these IMF components, the following two conditions need to be satisfied:

(1) In the entire time range, the number of local extremes and the number of points whose values are equal to zero must be equal or differ by at most one.

(2) At any moment, the mean value of the local maximum value of the upper envelope and the local minimum value of the lower envelope must be zero; that is, the upper envelope and the lower envelope are locally symmetric about the time axis.

2.2.2. Ensemble Empirical Mode Decomposition (EEMD). Although EMD decomposition can effectively divide the original signal into several IMF components for subsequent time–frequency analysis, it still has an obvious mode mixing problem, which means that an IMF component contains feature components with different time scales. One of the following situations is called mode mixing:

(1) In the same IMF component, there are different signals with a wide-scale distribution range.

(2) In different IMF components, there are signals with similar scales.

The mode mixing problem will make IMF lose single-scale features and then increase the difficulty of feature extraction and network training. Therefore, EEMD [28] is used to improve the mode mixing problem. Specifically, since the white noise has a mean value of zero, EEMD introduces uniformly distributed white noise many times in the process of signal EMD decomposition. As a result, the added white noise is able to mask the noise of the signal itself to obtain a more accurate upper and lower envelope. Finally, the decomposition result of EMD is averaged to reduce noise interference, and the mean value of the corresponding mode IMF components is taken as the EEMD decomposition results. In addition, the more the averaging number of processing, the less the noise affects the signal decomposition.

The decomposition results of EMD and EEMD are shown in Figure 3 (taking the signal of the last channel of the sixth group of signals in the dataset as an example). From Figure 3, it can be intuitively seen that for the same set of signals, the IMF components in the EEMD decomposition result are more symmetrical than those in the EMD, which is more in line with the requirements of signal decomposition.

2.2.3. Hilbert Transform. Let the original signal be $X(t)$, and after the EEMD decomposition, we obtain n IMF components and the residuals:

$$X(t) = \sum_{i=1}^{n} c_i + r_n. \tag{3}$$

For each IMF component $c_j(t)$, subject it to the Hilbert transform, i.e.,

$$H\left[c_j(t)\right] = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{c_j(t)}{t-\tau} d\tau. \tag{4}$$

The resolved signal $A\left[c_j(t)\right]$ and the parameters of the resolved signal in polar coordinates a and θ are as follows:

$$A\left[c_j(t)\right] = c_j(t) + iH\left[c_j(t)\right] = a_j(t)e^{i\theta_j(t)}, \tag{5}$$

$$a_j(t) = \sqrt{c_j^2(t) + H^2\left[c_j(t)\right]}, \tag{6}$$

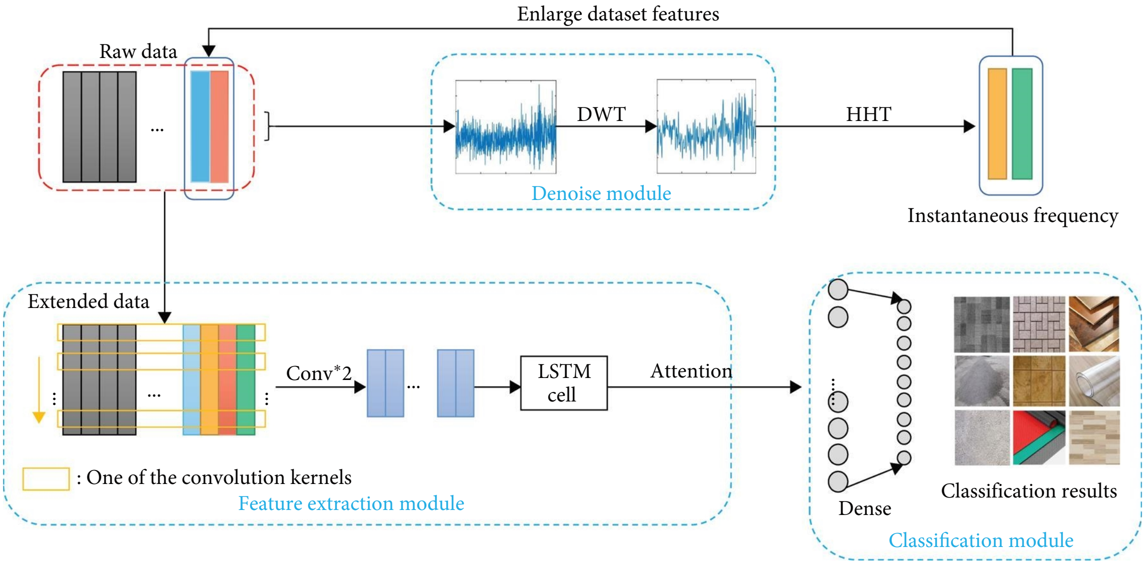

FIGURE 2: The model architecture diagram.

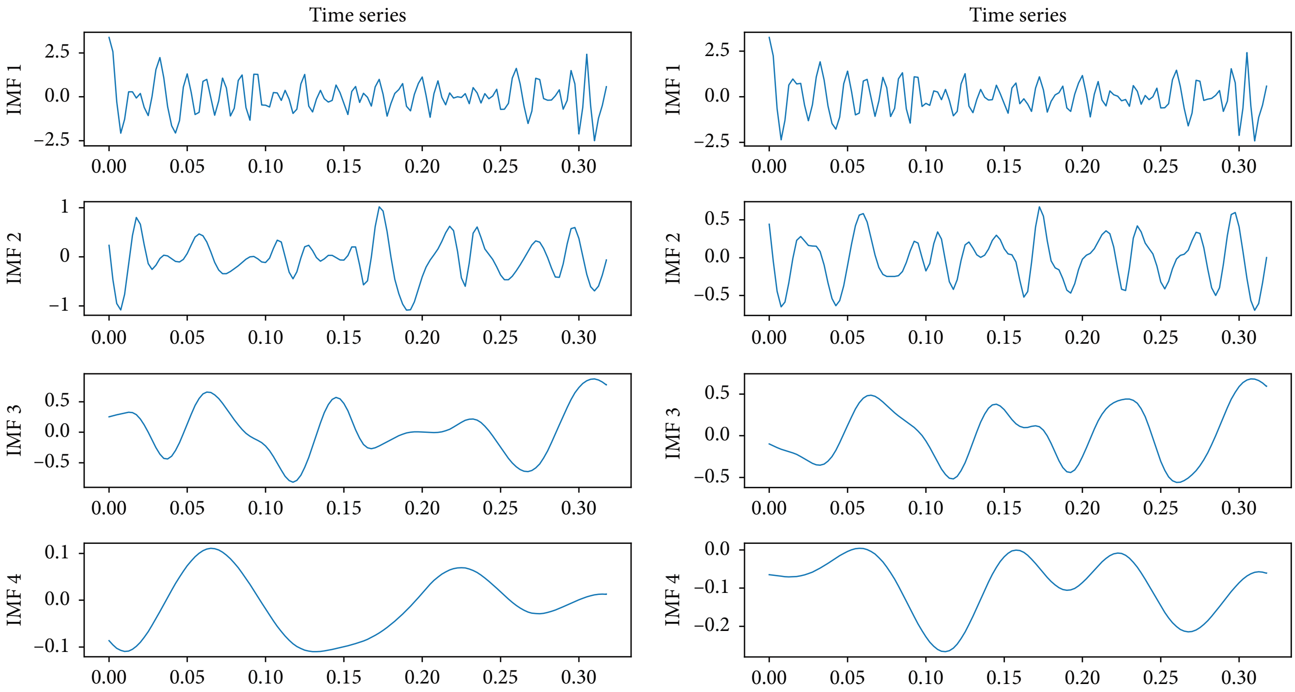

FIGURE 3: EMD and EEMD decomposition results for the signal in Figure 2 (limited by space, only the first four IMF components are shown; the left is the EMD decomposition result, and the right is the EEMD decomposition result).

$$\theta_j(t) = \arctan \frac{H\left[c_j(t)\right]}{c_j(t)}. \qquad (7)$$

The corresponding instantaneous frequency $f_j(t)$ and IMF components $c_j(t)$ are as follows:

$$f_j(t) = \frac{1}{2\pi}\frac{d\theta_j(t)}{dt}, \qquad (8)$$

$$c_j(t) = \mathrm{Re}\left(a_j(t)e^{i\theta_j(t)}\right) = \mathrm{Re}\left[a_j(t)\exp\left(i2\pi\int f_j(t)dt\right)\right]. \qquad (9)$$

As a result, we can obtain the instantaneous frequency information of each IMF component.

2.3. Neural Network

2.3.1. 1D CNN. Classical CNNs typically use 2D convolutional kernels, which perform convolutional operations

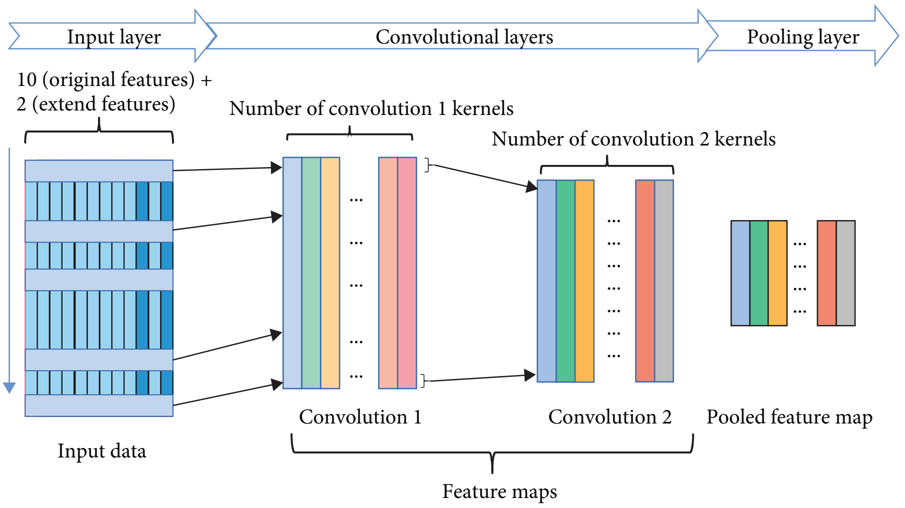

FIGURE 4: 1D convolutional neural network structure (in the convolutional layers, each color represents a set of convolutional kernels).

simultaneously in both the width and height directions. These networks are commonly used in computer vision and image processing [29]. 1D CNNs, on the other hand, are used for 1D input data and produce 1D output after convolution and pooling operations. They are primarily used in sequence modeling and natural language processing. Compared to 2D CNNs, 1D neural networks require fewer parameters, reducing their reliance on large-scale datasets.

For our task, we mainly use a 1D CNN to extract spatial features from the signal. It consists of two convolutional layers and a pooling layer, whose structure and operation are shown in Figure 4.

The convolutional layer is the core of the CNN, and its main role is to extract the feature information of the input. In the convolutional layer, the convolutional kernel performs convolutional operations on the output of the previous layer and outputs the convolutional result. Its mathematical model is as follows:

$$L_i^{l+1}(j) = K_i^l * x^l(j) + b_i^l, \tag{10}$$

where $L_i^{l+1}(j)$ denotes the convolution of the input $l+1$ layer. K_i^l denotes the weight of the ith first convolution kernel in layer l, $*$ denotes the convolution operation, $x^l(j)$ denotes the jth region in layer l, and b_i^l denotes the deviation of the ith convolution kernel in layer l.

The role of the pooling layer is to reduce the number of dimensions and parameters, generally divided into the maximum pooling method and average pooling method; in this task, we use the maximum pooling method, the function is as follows:

$$P_{i,m} = \max q_{i,(m-1)S+n}, \tag{11}$$

where $P_{i,m}$ denotes the ith feature map in the mth layer, $q_{i,(m-1)S+n}$ denotes the value of the $(m-1)S+n$ cell in the ith feature map, and S is the size of the overlap of adjacent sampling windows.

2.3.2. Long Short-Term Memory (LSTM). LSTM is a special form of RNN, which adds three gate structures on the basis of RNN, thus effectively solving the problem of gradient vanishing and gradient explosion in long sequence training [30]. Compared with ordinary RNN, LSTM can retain more historical information and avoid unnecessary interference when dealing with longer and more complex sequences, so it has better performance than RNN. In our task, each group of signals needs to collect 128 points, and the robot behavior has obvious temporal features, such as a robot performing different actions at different time periods. Therefore, we use LSTM to further extract the hidden temporal features in the signal after 1D CNN.

Suppose the current input of the LSTM is x^t, and the information passed down from the previous stage is h^{t-1}, then we have the following:

$$z = \tanh(WS), \tag{12}$$

$$z^i = \sigma(W^i S), \tag{13}$$

$$z^f = \sigma(W^f S), \tag{14}$$

$$z^o = \sigma(W^o S), \tag{15}$$

where S is a vector of x^t and h^{t-1} splices, $z^i z^f z^o$ are three gating signals that control the forget gate, the selective memory gate and the output gate respectively, and z are converted to a value between -1 and 1 by a $\tanh$ activation function.

As shown in Figure 5, there are three phases inside the LSTM during model work:

(1) Forgetting phase: selective forgetting of input from the previous node to control which of the previous state c^{t-1} needs to be retained and which needs to be forgotten by calculating the gated signal z^f.

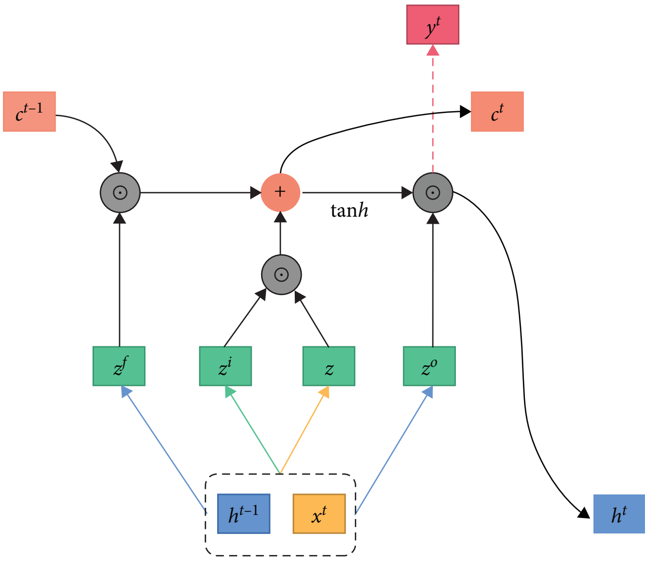

Figure 5: Structure of LSTM's internal work.

(2) Select memory phase: the input of the current phase x^t is selectively memorized, with the current input signal represented by the previously calculated z and z^i as the gated signal controlling this step.

From (1) and (2), we can calculate the transfer to the next state c^t as follows:

$$c^t = z^f \odot c^{t-1} + z^i \odot z, \quad (16)$$

where the ⊙ symbol represents matrix multiplication.

(3) Output phase: determines which output will be treated as the current state. This step is controlled primarily by z^o, while the previous stage c^o is scaled as follows:

$$h^t = z^o \odot \tanh(c^t). \quad (17)$$

The resulting output y^t is similar to the normal RNN and changes from h^t to the following:

$$y^t = \sigma(W'h^t). \quad (18)$$

2.3.3. Attention Mechanism. The internal mechanism of robot motion is complex, so there may be a lot of interference in signals collected through sensors. To prevent model interference, we introduced an attention layer after the LSTM layer to enable the model to capture more meaningful information to improve the accuracy of the final classification [31].

Assuming that the information from the LSTM layer is $H = [h_1, h_2, h_3 \ldots h_n]$, the attention's structure is shown as Figure 6.

(1) Calculate the correlation between the query vector q and each input h_i through the function $S(h, q)$, and in our work, select the dot product model as the S function in the following:

Table 1: Hyperparameter optimal combination.

Hyperparameter	Optimal value
Number of convolutional kernels	128
Size of convolutional kernel	3
Number of nodes of the LSTM layer	100
Time step of the LSTM	128
Probability values for the dropout layer	0.3
Learning rate	0.001

$$S(h, q) = h^T q. \quad (19)$$

(2) Use the softmax activation function to normalize these fractions, mapping to the 0–1 interval, and the result of normalization is the attention distribution of the query vector q on each input h_i,denoted as $a = [a_1, a_2, a_3 \ldots a_n]$, the h-array corresponds to the a-array, and the corresponding formula is as follows:

$$a_i = \frac{\exp(s(h_i, q))}{\sum_{j=1}^{N} \exp(s(h_j, q))}. \quad (20)$$

(3) According to the attentional distribution obtained in the previous step, the information can be extracted selectively from the input information, and the input information can be summarized and weighted according to the attentional distribution.

Context embodies the current focus of the model. The calculation formula is as follows:

$$\text{Context} = \sum_{i=1}^{n} a_i \cdot h_i. \quad (21)$$

(4) Finally, context is activated by the tanh function as an output of attention.

3. Experiments

3.1. Experiment Environment and Details. Our model is implemented in Tensorflow 2.11.0, and the training is carried out on a single NVIDIA Tesla T4. The training is done using the ADAM optimizer, and the batch size is set to 16. To ensure that the model could get the best possible classification effect, all cases of hyperparameter values were arranged and combined using classification accuracy as an indicator, which was identified using the 10-fold cross-validation method. Table 1 shows the optimal hyperparameter situation.

3.2. Dataset Introduction. The dataset used in the experiment was a public dataset [32] created in a Kaggle competition. The X_train dataset contains 3,810 groups of signal data collected by inertia measurement unit sensors. Each group of signals contains ten different channels and corresponds to one classification result, where each group of signals in each channel is 128 points accumulated by a single 400 Hz sensor over the same period. The signal waveforms of ten channels

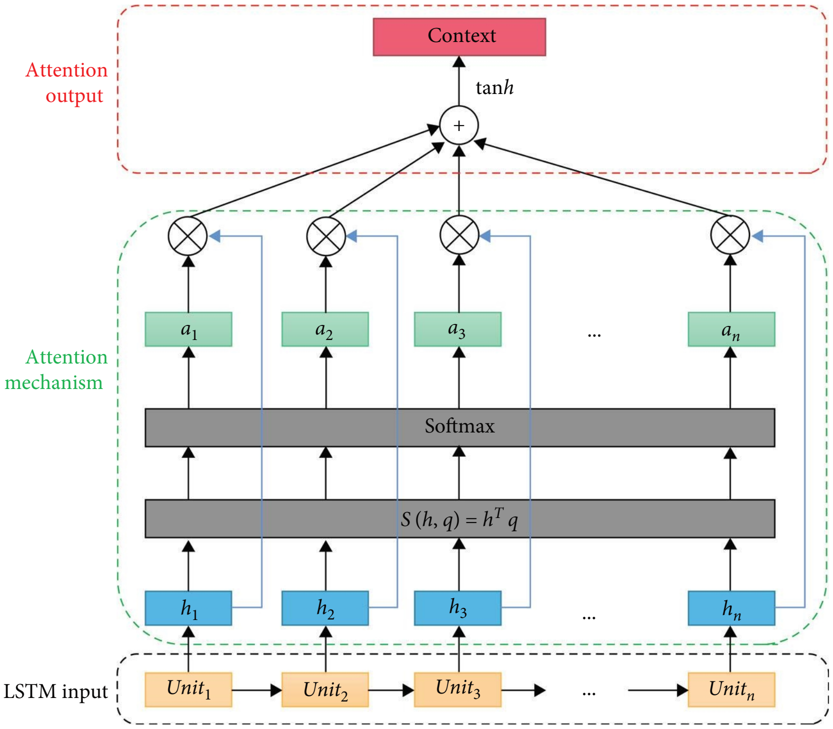

Figure 6: Structure of attention mechanism.

in a group of signals are shown in Figure 7. The Y_train dataset contains ground-type classification results for these 3,810 groups of signals, and the number of categories and corresponding categories are shown in Figure 8.

3.3. Experimental Process. In the experimental process, the DWT method was used to reduce the noise in the original dataset, giving the original data more distinct signaling characteristics.

In the second step, when the HHT was used to extract the frequency domain characteristics of the signal, the importance of 10 channels was sorted. The best combination of channels was selected through comparative experiments. The effects of the EMD and EEMD were also compared when decomposing the signal. After choosing the best channel combination, the signal data were decomposed using the EEMD method.

After that, each module of the proposed method was removed for the ablation experiment to prove the validity of the modules in the proposed method.

Finally, the technique used in this study was compared with some popular methods in this field to demonstrate its superiority.

3.3.1. DWT Denoise Comparison Experiment. First, the Daubechies 8 wavelet was selected as the wavelet basis function. We used the dwt_max_level function to adaptively select the level that best fits this signal decomposition. In the second step, six sets of high-frequency coefficient thresholds were selected to decompose the signal. Finally, the decomposed wavelet signal was reconstructed and trained on a base LSTM model with the same training parameters. The average classification accuracy on the test set in six cases was compared using a 10-fold cross-validation method. The results are shown in Table 2. It can be seen that the best classification effect is achieved when selecting a threshold of 0.06. Accordingly, a threshold of 0.06 was chosen for this study to reduce the noise of the original data, and the resulting new dataset was used as the input for the next step.

3.3.2. HHT Experiment. The dataset obtained in the previous step only reflects time domain features. Identifying hidden frequency domain information is significant to creating a high-performance network but cannot be explicitly extracted by the neural network. Therefore, a time–frequency analysis of the signal is needed. The HHT can solve the problem of unstable signals and obtain the instantaneous frequency of each moment signal more accurately. Therefore, HHT was used to extract instantaneous frequency information from the dataset.

To save computational costs and avoid interference between too many features, only the most essential channels in the frequency domain were analyzed for this study. For feature importance sorting, the permutation importance method was used. After classification accuracy was obtained on the model, the data and model parameters of the other channels were guaranteed to remain unchanged. Then, all the data obtained from a channel was randomly disrupted, and the classification test was reperformed. The importance of each channel is reflected by the difference in accuracy before and after the comparison.

Specifically, the first analysis performed was CNN-LSTM-ATTENTION, the network model in the proposed method, to test the initial accuracy. Then, the first through

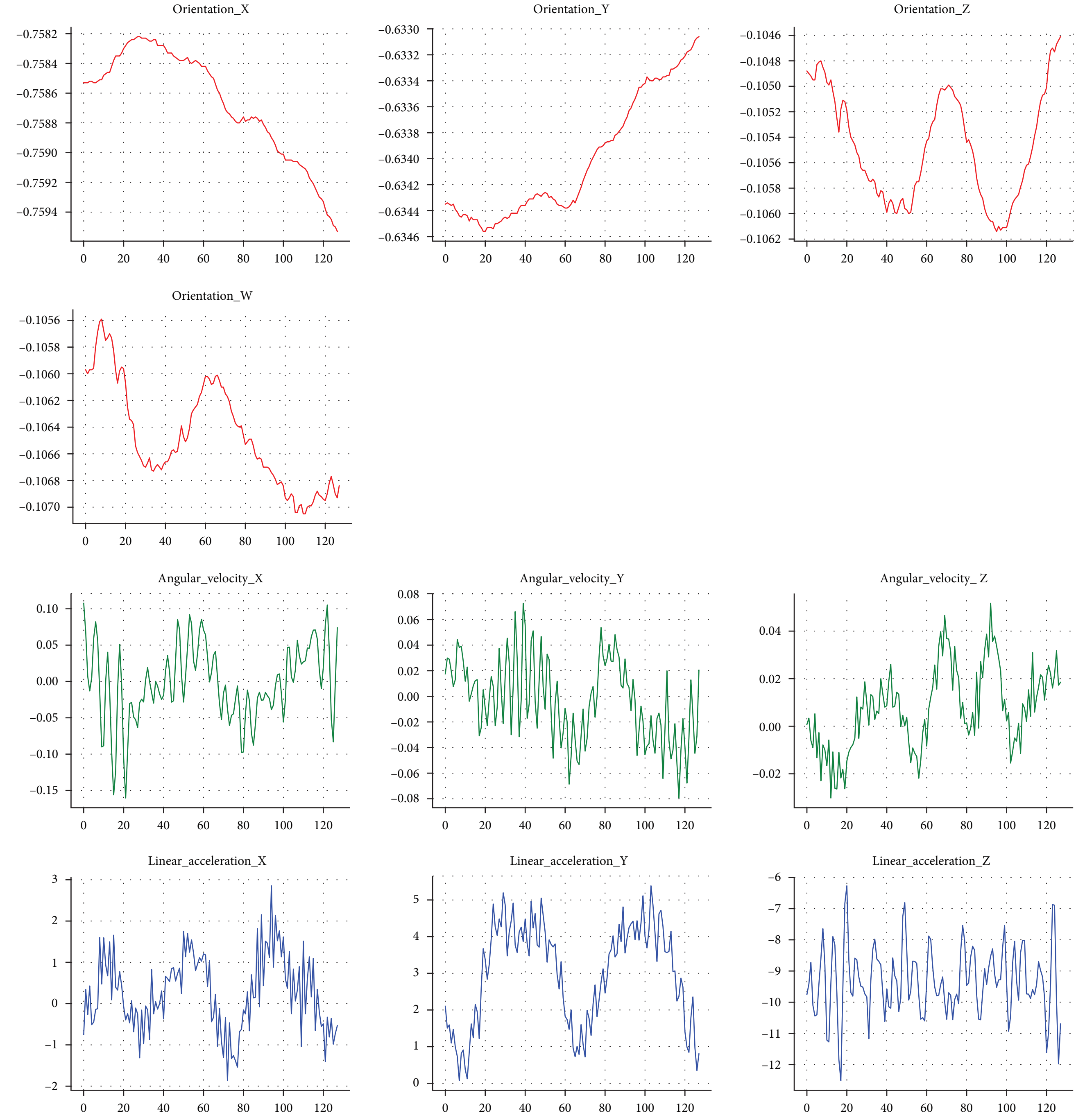

Figure 7: Visualization results of 10 signal waveforms corresponding to each ground media category (taking group 0 as an example).

tenth channels were disrupted. Five random disruptions were conducted to avoid disturbing the effects of randomness on experimental results, and an average was calculated and used as the result. The initial and disrupted accuracy rates are shown in Figure 9.

The length of the orange section in Figure 9 signifies characteristic importance and is sorted from large to small as follows: linear_acceleration_Z > linear_acceleration_Y > linear_acceleration_X > angular_velocity_Z > orientation_W > angular_velocity_X > angular_velocity_Y > orientation_X > orientation_Z > orientation_Y. Among these 10 features, the linear_acceleration_Z, linear_acceleration_Y, linear_acceleration_X columns are much more important than the other seven. Therefore, instantaneous frequency features from these three channels were added to the dataset to compare classification results.

For seven different combinations of these three channels, a comparative experiment was conducted to compare the added classification accuracy, as shown in Table 3. As can be seen from Table 3, the fusion of the two most essential channels, linear_acceleration_Y, and linear_acceleration_Z, yields the highest classification result, which aligns with the

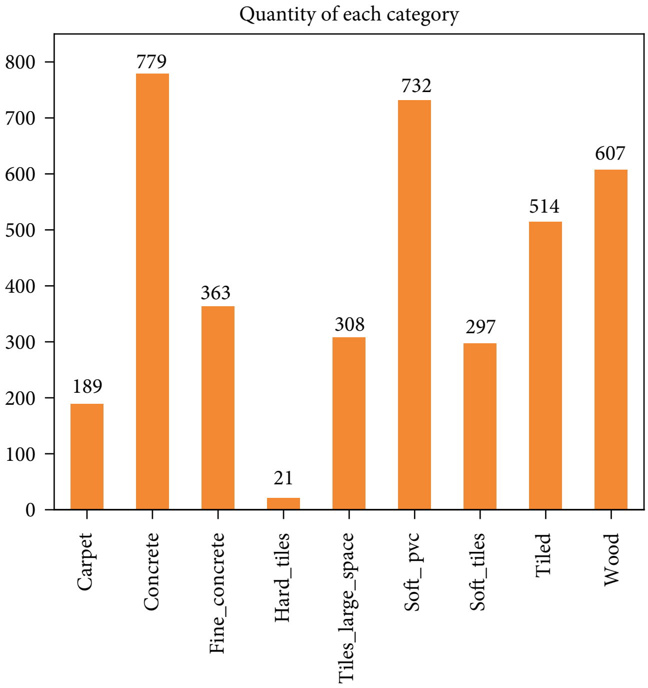

Figure 8: Visualization of nine different types of ground media and their quantities.

Table 2: Classification accuracy corresponding to different denoise thresholds in DWT.

Denoise threshold	Average accuracy (%)
Original dataset	73.02
Universal threshold	68.23
0.04	73.99
0.06	74.08
0.07	74.04
0.08	71.72
0.1	73.29

expectations of this study. However, there are some cases where the classification results are not as primitive. For example, there are several combinations that yield even worse results than not adding features. We hypothesized that the decomposition of the EMD in the traditional HHT method would lead to mode mixing [28], resulting in a significant error in the decomposed IMF components, which interferes with the learning of the model. Because of this, an EMD derivative optimization method, EEMD, was used to redecompose the signal and compare it to EMD.

In our experiments, we chose to extract the last-order IMF component of the EEMD decomposition for the following reasons:

On the one hand, since our algorithm targets a small dataset, we need to avoid extracting too many numbers of IMF components, so for the two columns of signals with the highest importance, we extract only one IMF component each.

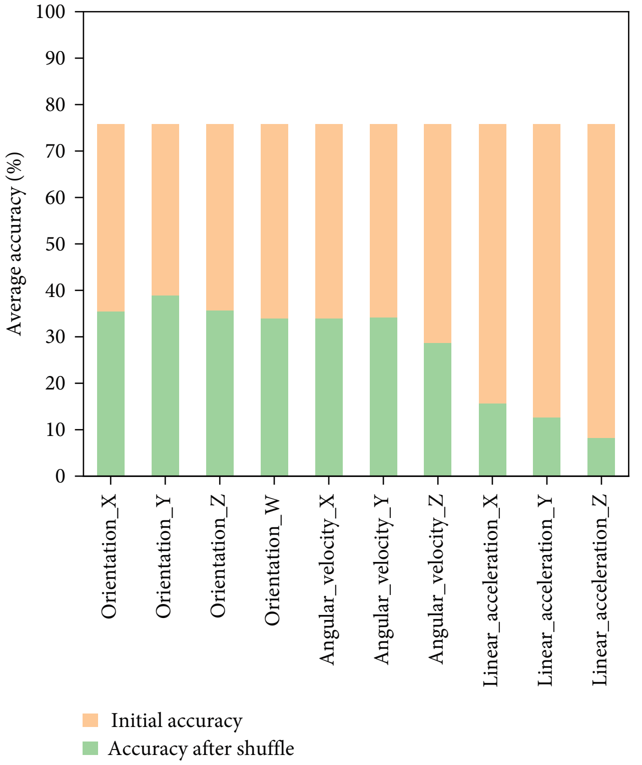

Figure 9: Channel importance sorting results.

Table 3: Accuracy of different channel combinations (the channel in the channel combination is meant to expand the dataset by extracting instantaneous frequency features from the channel using HHT. If there is more than one channel, it means that their instantaneous frequency feature are extracted channel by channel and added to the original signal dataset).

Channel combinations	Classification accuracy (%)
Linear_acceleration_X	76.90
Linear_acceleration_Y	74.02
Linear_acceleration_Z	74.28
Linear_acceleration_X, Y, Z	75.85
Linear_acceleration_X, Y	74.28
Linear_acceleration_X, Z	71.13
Linear_acceleration_Y, Z	78.74
No features added	76.11

On the other hand, the information with high frequency appears relatively short and rapid, so the LSTM model can effectively remember this information; for the information with lower frequency, the overall time interval is relatively long, and the LSTM may forget in the long period of memory, so we extract the lowest frequency IMF component to make up for this deficiency of the LSTM, which helps the model to remember long-term information more effectively and improves the learning ability of the model.

We ensured that other conditions remained unchanged and added EEMD-extracted instantaneous frequency features to the dataset. These results were then compared with

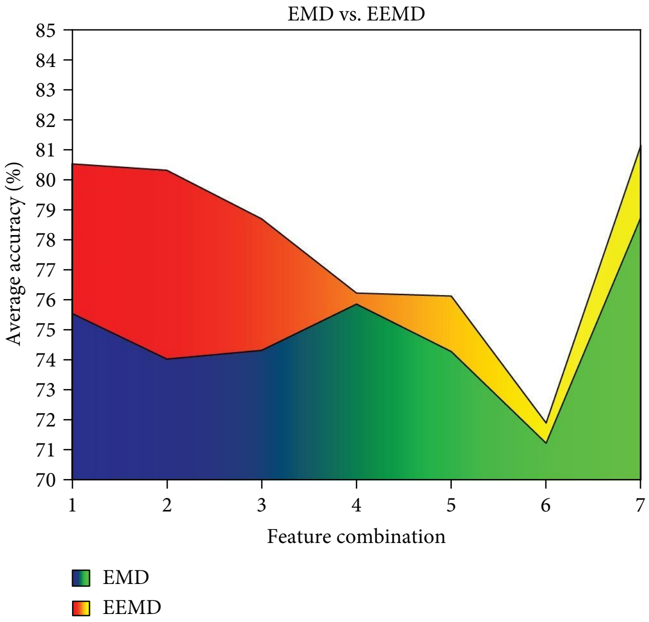

FIGURE 10: Comparison results of EMD and EEMD (the horizontal coordinates 1–7 represent the first seven different combinations of channels in Table 3, and the vertical coordinates are the average accuracy of the classification).

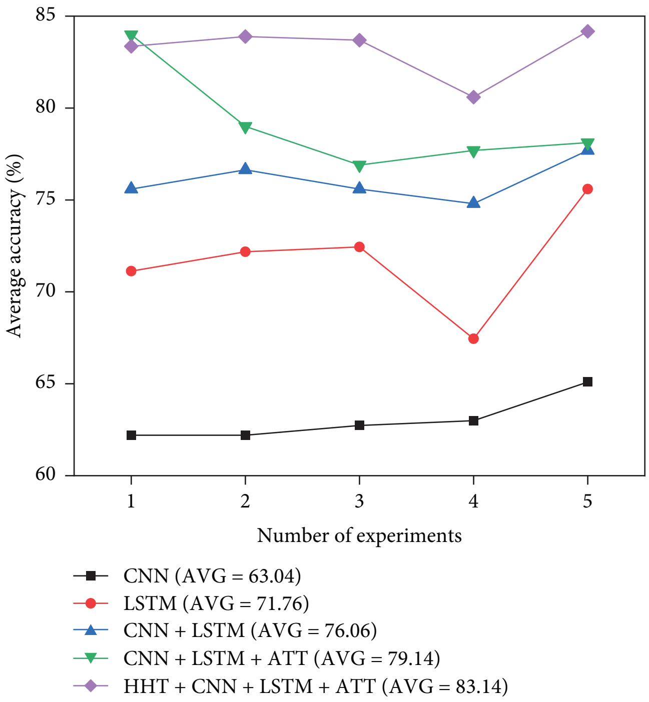

FIGURE 11: Ablation experiment results.

the EMD method, as shown in Figure 10, where the transverse coordinates represent the first seven combinations from 1 to 7. The ordinate represents the EMD and EEMD results for each experiment.

Figure 10 shows that the accuracy of the EEMD decomposition is significantly improved compared to the EMD decomposition under seven different experimental conditions, and the best combination of channels is still a combination of linear_acceleration_Y and linear_acceleration_Z. Therefore, this study adopted the HHT method of EEMD decomposition to conduct subsequent experiments on the instantaneous frequency extraction and expansion dataset of these two channels.

3.3.3. Ablation Experiment. To verify the effectiveness of the combined model, feature extraction, and fusion of the HHT, the following best combination of modules, with other conditions remaining the same, were chosen: experiments with a separate CNN classification, separate LSTM classification, combined CNN-LSTM classification, combined CNN-LSTM-ATTENTION classification and combined HHT-CNN-LSTM-ATTENTION classification. Each group of experiments was conducted five times, and the average classification accuracy was taken as the evaluation index.

Only the spatial features between the different sensors were extracted when the CNN was classified separately. The LSTM classification only extracted the timing features of each sensor at different moments. Therefore they both have low classification accuracy. The results show that the accuracy of the CNN-LSTM classification is higher than that of the CNN and LSTM classifications alone. As can be seen from Figure 11, after the attention mechanism was added, the model's efficiency was improved, and the model could focus on further analysis of the signal. Further, after extracting the frequency feature with the HHT, the time domain and frequency domain information could be fuzed to analyze the signal from multiple angles and provide more helpful information to the model. Therefore, classification accuracy was further increased, and the experimental results aligned with the expected results.

In addition, to more clearly determine whether the model relies more on HHT or attention, we also performed a combination of CNN-HHT (AVG = 68.26), LSTM-HHT (AVG = 74.41), and CNN-LSTM-HHT (AVG = 81.33). Therefore, we can see that HHT has a greater improvement for the model than attention, and the model relies more on HHT, which again proves the superiority of our proposed method.

3.3.4. Comparison with Other Common Methods. To test the performance of the proposed method, four common methods in this field were selected to compare results. The classification results are shown in Table 4. As can be seen from Table 4, our inference time is indeed longer than the remaining methods, but the feature extraction part of the remaining methods takes longer, and the overall computational cost is larger. The key point is that our method has the highest accuracy, so we think that our method is the best.

4. Conclusion

Using the open dataset from a Kaggle competition as the research object, this paper studies the problem of robot ground-type classification based on HHT and attention-based spatiotemporal coupling networks. First, the DWT was used to reduce the noise of the original signal. Then, a denoise threshold of 0.06 was selected based on classification

Table 4: Comparison of our proposed method with other common methods in the field (HHT spectrograms refers to the CNN-based of HHT spectrograms image classification method).

Other common methods	Accuracy (%)	Inference time (ms)
PSO-SVM	61.07	37
PCASVM	65.11	39
DWT + Random forest	80.74	35
HHT spectrograms	71.28	72
Ours model	83.14	66

accuracy under the same conditions. In the HHT experiment, the importance of all channels in the dataset was sequentially sorted using the permutation importance method. The linear_acceleration_X, linear_acceleration_Y, and linear_acceleration_Z were found to have the most significant impact on the results. Therefore, seven different combinations of these three channels were compared. It was found that the best classification effect can be obtained by combining the linear_acceleration_Y and linear_acceleration_Z channels. After that, a comparative analysis of EMD and EEMD was conducted. It was determined that EEMD is superior to EMD in five experimental verifications. Thus, the EEMD decomposition for linear_acceleration_Y and linear_acceleration_Z was used. Next, the last IMF component in EEMD was selected to perform the Hilbert transform to obtain the instantaneous frequency to enlarge the original dataset features. After passing through two 1D CNN layers, LSTM, attention structures, and a fully connected layer, the expanded dataset obtained an average classification accuracy of 83.14%. Finally, ablation experiments and contrast experiments with other common methods were conducted in turn. It was determined that the classification accuracy increased after each module was added. It was also proven that the method utilized in this study is superior to other common methods in the field, demonstrating the validity of each module in this study's proposed method and the superiority of the overall approach.

The authors of this study believe that future studies in this field should go in the following directions:

(1) When reconstructing a signal after EEMD decomposition, it is necessary to select which IMF components to reconstruct and to determine which IMF components should be used under what conditions. Presently, this research aspect lacks theoretical support and interpretability, and IMF components can only be selected using comparative experiments and experiences.

(2) When using a neural network for classification, if the distribution of categories in the results is uneven, the classification accuracy of the minority class samples will be lower. How to solve this imbalance is a direction for future research.

Data Availability

Data Source: https://www.kaggle.com/competitions/career-con-2019/data.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] P. Papadakis, "Terrain traversability analysis methods for unmanned ground vehicles: a survey," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 4, pp. 1373–1385, 2013.

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

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

[4] M. Ono, T. J. Fuchs, A. Steffy, M. Maimone, and J. Yen, "Risk-aware planetary rover operation: autonomous terrain classification and path planning," in *2015 IEEE Aerospace Conference*, pp. 1–10, IEEE, 2015.

[5] K. Walas, "Terrain classification and negotiation with a walking robot," *Journal of Intelligent & Robotic Systems*, vol. 78, pp. 401–423, 2015.

[6] A. Howard and H. Seraji, "Vision-based terrain characterization and traversability assessment," *Journal of Robotic Systems*, vol. 18, no. 10, pp. 577–587, 2001.

[7] J.-F. Lalonde, N. Vandapel, D. F. Huber, and M. Hebert, "Natural terrain classification using three-dimensional ladar data for ground robot mobility," *Journal of Field Robotics*, vol. 23, no. 10, pp. 839–861, 2006.

[8] K. Iagnemma, S. Kang, H. Shibly, and S. Dubowsky, "Online terrain parameter estimation for wheeled mobile robots with application to planetary rovers," *IEEE Transactions on Robotics*, vol. 20, no. 5, pp. 921–927, 2004.

[9] M. B. Alatise and G. P. Hancke, "A review on challenges of autonomous mobile robot and sensor fusion methods," *IEEE Access*, vol. 8, pp. 39830–39846, 2020.

[10] K. Zhao, M. Dong, and L. Gu, "A new terrain classification framework using proprioceptive sensors for mobile robots," *Mathematical Problems in Engineering*, vol. 2017, 14 pages, 2017.

[11] X. A. Wu, T. M. Huh, R. Mukherjee, and M. Cutkosky, "Integrated ground reaction force sensing and terrain classification for small legged robots," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 1125–1132, 2016.

[12] F. L. Garcia Bermudez, R. C. Julian, D. W. Haldane, P. Abbeel, and R. S. Fearing, "Performance analysis and terrain classification for a legged robot over rough terrain," in *IEEE International Workshop on Intelligent Robots and Systems*, pp. 513–519, IEEE, 2012.

[13] J. Christie and N. Kottege, "Acoustics based terrain classification for legged robots," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3596–3603, IEEE, 2016.

[14] Y. Yang, Z. Peng, W. Zhang, and G. Meng, "Parameterised time-frequency analysis methods and their engineering applications: a review of recent advances," *Mechanical Systems and Signal Processing*, vol. 119, pp. 182–221, 2019.

[15] N. Hazarika, J. Z. Chen, A. C. Tsoi, and A. Sergejew, "Classification of EEG signals using the wavelet transform," *Signal Processing*, vol. 59, no. 1, pp. 61–72, 1997.

[16] N. Saravanan and K. I. Ramachandran, “Incipient gear box fault diagnosis using discrete wavelet transform (DWT) for feature extraction and classification using artificial neural network (ANN),” *Expert Systems with Applications*, vol. 37, no. 6, pp. 4168–4181, 2010.

[17] S. A. Deokar and L. M. Waghmare, “Integrated DWT–FFT approach for detection and classification of power quality disturbances,” *International Journal of Electrical Power & Energy Systems*, vol. 61, pp. 594–605, 2014.

[18] N. E. Huang, Z. Shen, S. R. Long et al., “The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis,” *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 454, no. 1971, pp. 903–995, 1998.

[19] K. Fu, J. Qu, Y. Chai, and Y. Dong, “Classification of seizure based on the time-frequency image of EEG signals using HHT and SVM,” *Biomedical Signal Processing and Control*, vol. 13, pp. 15–22, 2014.

[20] M.-F. Guo, N.-C. Yang, and W.-F. Chen, “Deep-learning-based fault classification using Hilbert–Huang transform and convolutional neural network in power distribution systems,” *IEEE Sensors Journal*, vol. 19, no. 16, pp. 6905–6913, 2019.

[21] A. Subasi, “Classification of EMG signals using PSO optimized SVM for diagnosis of neuromuscular disorders,” *Computers in Biology and Medicine*, vol. 43, no. 5, pp. 576–586, 2013.

[22] X. Du and H. Zhu, “Track robot based on time-frequency characteristics and PCA-SVM ground classification study,” *Journal of Henan University of Technology*, pp. 84–90, 2019.

[23] E. Gokgoz and A. Subasi, “Comparison of decision tree algorithms for EMG signal classification using DWT,” *Biomedical Signal Processing and Control*, vol. 18, pp. 138–144, 2015.

[24] S. S. Baishya and B. Bauml, “Robust material classification with a tactile skin using deep learning,” *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 8–15, 2016.

[25] N. Duan, L.-Z. Liu, X.-J. Yu, Q. Li, and S.-C. Yeh, “Classification of multichannel surface-electromyography signals based on convolutional neural networks,” *Journal of Industrial Information Integration*, vol. 15, pp. 201–206, 2019.

[26] S. K. Prabhakar, H. Rajaguru, and S.-W. Lee, “A framework for schizophrenia EEG signal classification with nature inspired optimization algorithms,” *IEEE Access*, vol. 8, pp. 39875–39897, 2020.

[27] M. Liu, G. Liao, N. Zhao, H. Song, and F. Gong, “Data-driven deep learning for signal classification in industrial cognitive radio networks,” *IEEE Transactions on Industrial Informatics*, vol. 17, no. 5, pp. 3412–3421, 2021.

[28] Z. Wu and N. E. Huang, “Ensemble empirical mode decomposition: a noise-assisted data analysis method,” *Advances in Adaptive Data Analysis*, vol. 1, no. 1, pp. 1–41, 2009.

[29] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, “Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising,” *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017.

[30] A. Sherstinsky, “Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network,” *Physica D: Nonlinear Phenomena*, vol. 404, Article ID 132306, 2020.

[31] Z. Niu, G. Zhong, and H. Yu, “A review on the attention mechanism of deep learning,” *Neurocomputing*, vol. 452, pp. 48–62, 2021.

[32] “CareerCon 2019 - Help Navigate Robots,” 2019, https://www.kaggle.com/competitions/career-con-2019/data.