IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 15, 2019, accepted September 11, 2019, date of publication September 17, 2019, date of current version October 3, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2941977

Movements Classification of Multi-Channel sEMG Based on CNN and Stacking Ensemble Learning

SHU SHEN[1,2], (Member, IEEE), KANG GU[1], XIN-RONG CHEN[3,4], MING YANG[1], AND RU-CHUAN WANG[1,2]

[1]School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China
[2]Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks, Nanjing 210023, China
[3]Academy for Engineering and Technology, Fudan University, Shanghai 200433, China
[4]Shanghai Key Laboratory of Medical Image Computing and Computer Assisted Intervention, Shanghai 200032, China

Corresponding author: Xin-Rong Chen (chenxinrong@fudan.edu.cn)

This work was supported in part by the National Science Foundation of China under Grant 61401221, Grant 61873131, Grant 61872196, Grant 61701168, Grant 61572261, and Grant 61572260; in part by the China Postdoctoral Science Foundation under Grant 2016M601860; in part by the Postgraduate Research and Practice Innovation Program of Jiangsu Province under Grant SJCX19_0240 and Grant SJKY19_0823; in part by the NJUPT Teaching Reform Project under Grant JG00417JX74; and in part by the Teaching Reform from Project of Tongda College, NJUPT, under Grant JG30618003.

ABSTRACT In recent years, the analysis of surface electromyography (sEMG) signals by feature engineering and machine learning has developed rapidly. However, when feature engineering is applied to feature extraction of sEMG signals, important feature information in the signals will inevitably be omitted, which will reduce the performance of signal analysis and recognition. Therefore, this paper proposes a method to complete classification of sEMG hand movements based on convolutional neural network (CNN) and stacking ensemble learning. In this method, a primary classifier based on CNN is designed to extract sEMG data features, which avoid omission of important feature information. A secondary classifier based on the stacking method is designed to integrate three primary classifiers trained with time domain, frequency domain and time-frequency domain data of the sEMG signal respectively. Then, several experiments on NinaPro DB5 dataset is performed to evaluate the proposed models. When the window length is 200ms, primary classifier is trained and tested with the sEMG signal data divided by the 80ms, 100ms, and 125ms sliding length. The best accuracy can reach 71%. The primary classifier and the secondary classifier trained and tested with sEMG signal data divided by window lengths of 200ms and 300ms in the case of a sliding length of 100ms. When the window length is 200ms, the best primary classifier accuracy and the best secondary classifier accuracy can be 70.92% and 72.09%, respectively. On the window length of 300ms, the best primary classifier accuracy and the best secondary classifier accuracy can reach 75.02% and 76.02%, respectively. Finally, the model designed is compared with Linear Discriminant Analysis (LDA), Long Short Term Memory-CNN (LCNN), Support Vector Machine (SVM), and Random Forests. Under the same conditions, the average accuracy of the secondary classifier is 11.5%, 13.6%. and 10.1% higher than LDA, SVM, and LCNN, respectively. Also, the average accuracy rate is 3.05% higher than SVM and Random Forests.

INDEX TERMS Surface electromyography, movements classification, convolutional neural network, ensemble learning.

I. INTRODUCTION

Electromyography (EMG) is a superposition of bioelectrical signals generated by the muscles of the human body. sEMG is a way to detect muscle activity from the surface of human skin [1]. A considerable amount of studies have shown that sEMG can be used to detect the motional intention of the body. A number of analytical systems have been developed to understand human intent from sEMG signals and are used in non-invasive human-computer interaction systems such as hand prosthetic control, wheelchair control, exoskeleton, and virtual interaction [2]. With the maturity of sEMG signal acquisition technology, the analysis and recognition of sEMG signal has attracted the attention of researchers.

The associate editor coordinating the review of this manuscript and approving it for publication was Qingxue Zhang.

The advantage of using sEMG for movements recognition is that we can apply electrodes to any muscle area without too many restrictions and can be applied to disabled people with amputation. The disadvantage is that the identification is more difficult and not very stable. However, we firmly believe that deep learning can solve this problem.

At present, the feature engineering and machine learning methods are used to analyze and identify sEMG signals. Meattini *et al.* [3] proposed a robotic arm control system based on eight fully differential sEMG sensors. The system uses the surface electrode to collect the original sEMG of the human body. It combines the embedded system to filter the original signal, and uses the sliding window technology to extract Root Mean Square (RMS) characteristic of the signal. The feature data was trained and identified using SVM Model, and 90% accuracy was achieved in the robotic gripping task. Jose *et al.* [4] used multiple sEMG sensors to acquire sEMG signals from the biceps and triceps at a sampling frequency of 10kHz, and selectively extracted integral electromyogram (iEMG) and Slope Sign Changes (SSC). Multi-layer perceptron neural network and random forest model are used to recognize the internal and external rotation of the arm, and the accuracy is 91.6% and 97.7% respectively. As above, a bit feature of sEMG data was applied to identify fewer movements by using maching learning model.

There are also many studies on a large number of hand movements recognition methods based on sEMG. Kuzborskij *et al.* [5] analyzed and identified the sEMG signal of the 52 hand movements in the NinaPro dataset. They extracted various features of the multiple signals, including Mean Absolute Value (MAV), histogram (HIST), Multidimensional Discrete Wavelet Transform (mDWT), and Short-Time Fourier Transform (STFT), and used multiple models for training and evaluation, for example, Nonlinear Support Vector Machines (SVM-BRF), Multilayer Perceptron (MLP), k Nearest Neighbor (k-NN), Linear Support Vector Machine (SVM-linear), and LDA models. In the experiment, they found that the correct recognition rate of 52 types of hand motion can reach about 80% by combing MAV features and SVM-BRF model.

Gijsberts *et al.* [6] classified the 6 DOF force activation in the NinaPro dataset and the sEMG signals of 40 discrete hand movements. They used RMS, HIST, mDWT and Kernel Rule Least Squares (KRLS) algorithm to classify hand movements. The accuracy of predicting 6-DOF force activation was about 90% in 40 subjects and 80% in 40 discrete gesture classifications. Pizzolato *et al.* [7] analyzed the sEMG signal data in the NinaPro DB1, DB2, DB4, DB5. The sEMG signals in these data sets are collected by four different acquisition devices. The author compared the sEMG signal data from the time domain and the frequency domain, and expounds their respective characteristics. The RMS, TD, HIST, and mDWT features and random forest and SVM model were used to classified 41 gestures. The results showed that the optimal accuracy can reach about 74%.

The sEMG signal recognition method based on feature engineering and machine learning has a great dependence on the quality of feature engineering. It requires considerable consumption to extract and test features. In recent years, deep learning, a new machine learning technology, is beginning to be used for the analysis and recognition of sEMG signals. Its motivation lies in the establishment and simulation of the neural network for human brain analysis and learning. When Deep Neural Network (DNN) used to extract the features of sEMG signals, it can effectively avoid missing the effective information in signals and improve the accuracy of recognition. Xing *et al.* [8] used the CNN model with five convolutional layer parallel architectures to extract and classify sEMG signals. The accuracy rate is 5.71% higher than the random forest model and 4.06% higher than the support vector machine model. Atzori *et al.* [9] used a convolutional network to classify an average of 50 hand movements of 67 intact subjects and 11 transradial amputees. The results show that the recognition accuracy is higher than the traditional machine learning method. X. Zhai et al. [10] proposed a self-recalibrating classifier. It can automatically calibrates the original classifier. The results showed that the accuracy of calibrated classifier would be 10.18% (intact, 50 movement types) and 2.99% (amputee, 10 movement types) higher than uncalibrated classifier.

He *et al.* [11] combined long-short-term memory networks and multi-layer perceptrons to classify sEMG signal in NinaPro DB1 dataset. Approximately 75% accuracy was achieved in the classification of 52 types of hand movements in 27 subjects. Hu *et al.* [12] proposed a hybrid CNN and RNN model based on attention, which was tested on the databases, which are NinaProDB1, NinaProDB2, BioPatRec subdatabase, CapgMyo subdatabase and csl-hdemg database. The accuracy of the model were 87.0%, 82.2%, 94.1%, 99.7% and 94.5%, respectively. And they were 9.2%, 3.5%, 1.2%, 0.2% and 5.2% higher than the state at that time. Y.Wu et al. [13] proposed the LCNN model. The pre-processed sEMG signal can directly be input into the network for the dynamic recognition of gestures. The evaluation is based on NinaPro DB5 using standard test method, and the accuracy of Exercise A and Exercise B on gesture data was 71.66% and 61.4%, respectively.

As stated above, it is obviously that the deep learning method can overcome the limitation of the feature engineering requiring better feature quality. And many works have shown that the accuracy of classifying sEMG signals using DNN is generally higher. However, at this stage, the sEMG signal recognition based on deep learning model is hopeful to be improved in terms of the accuracy and the feature engineering complexity.

In this paper, we propose a multi-channel sEMG signal recognition model based on convolutional network (CNN) and Stacking ensemble learning. In the model, the primary classifier is formed using the Sequeeze-and-Excitation Block, Inception Block and fully connected network, in which the features of sEMG signals are extracted using CNN to avoid missing important feature information. Three different primary classifiers are trained using the representation data of the time domain, frequency domain and time frequency domain of the sEMG signal, respectively. The secondary classifier uses the Stacking method to generate the final classification result by integrating three different primary classifiers, which can further improve the classification accuracy. In order to verify the performance of the model, we compare the model designed with the current mainstream methods on the classification accuracy.

This article is divided into four main parts. Section I mainly introduces the research background and related work on sEMG movements recognition. Multi-channel sEMG movements recognition model we designed is described in section II. Section III verifies the performance of the model applied on the NinaPro DB5 data set. Section IV summarizes this paper.

II. METHODS

A. FRAMEWORK

The framework of the proposed multi-channel sEMG movements recognition is shown in Fig. 1 This framework is divided into the DATA PROCESSING module and the CLASSIFIER module. In the DATA PROCESSING section, the sEMG is transformed and normalized to the time domain, frequency domain and time-frequency domain data of the signal. The CLASSIFIER section is composed of Primary classifier and Secondary classifier. The primary classifier based on CNN is mainly used for the feature extraction and the movements classification of sEMG signals. The secondary classifier integrate three primary classifiers through Stacking method, which is trained by time domain, frequency domain and time-frequency domain data. The classification results of the three primary classifiers are combined and applied as feature vectors to train the secondary classifier. These modules will be described in detail as follows.

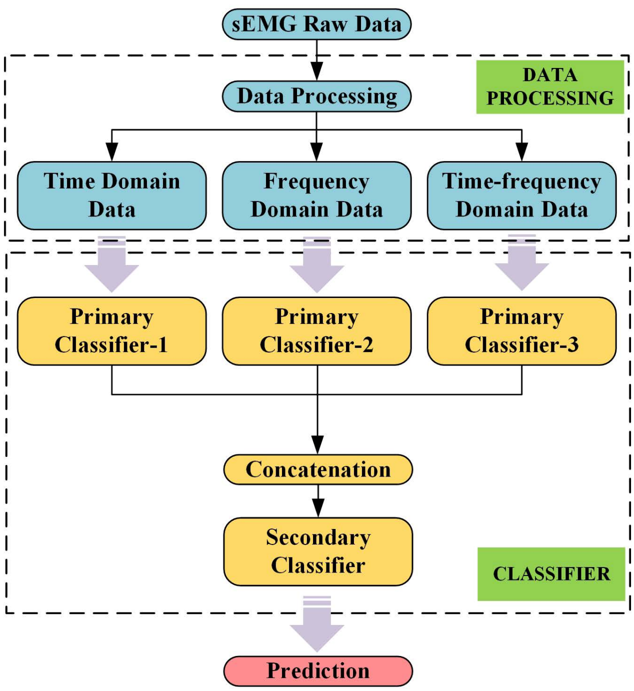


FIGURE 1. Framework of multi-channel sEMG movements recognition.

B. DATA PROCESSING

Generally, sEMG is processed through sliding windows. For real-time prosthetic control using sEMG, input delay is an important factor to be considered. Hudgins *et al.* [14] proposed a maximum allowable delay of 300 ms. In this paper, we select the sliding window of 200ms and the incremental window of 100ms to divide the sEMG signal of each channel into a window sequence. Fig. 2 shows how to segment single-channel sEMG signal by sliding window.

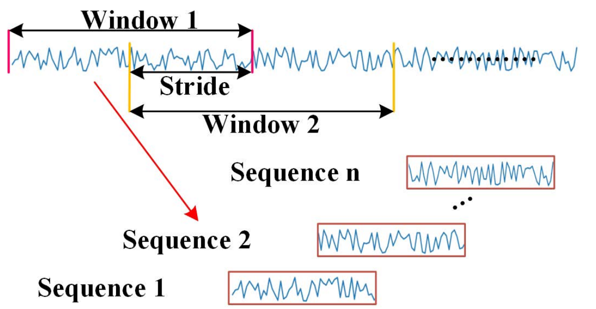


FIGURE 2. Single channel signal windowing.

The sEMG signal data segmented can meet the requirements of real-time recognition. And to satisfy the requirements of model input, the data is aligned to a fixed length. Next, in order to train the three primary classifiers, we perform Discrete Fourier Transform (DFT) and Discrete Wavelet Packet Transform (DWPT) on the windowed sEMG signal to obtain the frequency domain representation and the time-frequency domain representation of the sEMG.

When performing DFT, assuming that the sequence length of the current sEMG signal is N, we extend the length of the sequence to $2N$ by adding 0 at the end.The extended sequence is subjected to DFT, and the first N amplitude data of the transformed sequence is taken as a frequency domain representation of the signal.

Discrete Wavelet Packet Transform (DWPT) is an extension of Discrete Wavelet Transform (DWT), which can perform time-frequency analysis and can analyze the low frequency and high frequency coefficients of the signal [15], [16]. Fig. 3 shows the decomposition process of DWPT for sEMG. Low-pass and high-pass filtering are performed on the sEMG signals with the frequency range of $0 \sim F$ Hz. The low-pass and high-pass filtering of the sEMG signal is downsampled, and the signal subsequence of the first-order decomposition is obtained. Then, the subsequence generated can be further decomposed for n times. And the subsequences of different frequency bands are generated

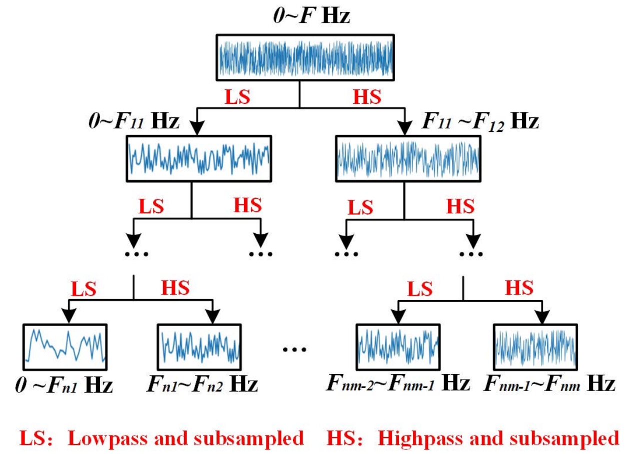


FIGURE 3. DWPT on windowed sEMG.

by the n-level wavelet packet transform. The wavelet type used for wavelet packet decomposition and the number of decomposition stages can be selected arbitrarily. In this paper, we apply db7 wavelet and performed level 2 decomposition.

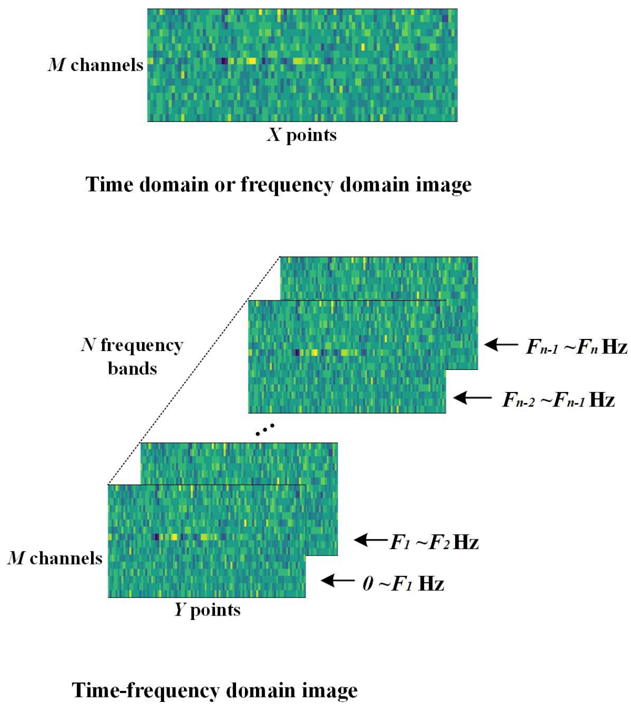


FIGURE 4. sEMG image representation method.

Through the above processing, the sEMG signal data are divided into time domain data, frequency domain data and time-frequency domain data. Since the sEMG data of different systems are collected by different analog-to-digital converters, their values range is different. we performed min-max normalization on the time domain, frequency domain and time-frequency domain of sEMG. After the normalization, we represent the time domain, frequency domain and time-frequency domain data by images shown in Fig. 4. The time domain or frequency domain data of the sEMG can be represented as an image of length X, width M, and channel number 1, where M is the number of acquisition channels of sEMG, and X is the number of sampling points of each subsequence after the segmentation of sEMG signal data by windowing. The time-frequency domain data of the sEMG can be represented as an image of length Y, width M, and channel number N, where M is also the number of acquisition channels of sEMG. And Y is the length of each subsequence after wavelet packet decomposition for sEMG which has been segmented by the windowing method. N is the number of subsequences obtained after wavelet decomposition. These subsequences belong to different frequency bands, so they are placed in different channels of the image.

C. CNN BASED PRIMARY CLASSIFIER

Inception Block is an important module of the primary classifier and its structure is shown in Fig. 5. In module firstly, three different types of convolution operations are used to extract the different scale features of the input data. Then, a convolution operation is performed on the connected multi-scale feature data for the purpose of feature fusion and dimensionality reduction. If the original data $[H, W, 1]$ with the length H, the width W, and the number of channel 1, the scale of each type of feature data is $[H, W, \mathit{Filters}]$ after performing three types of convolution operations. Each type of feature data is connected on the channel to obtain feature data with $[H, W, 3{*}\mathit{Filters}]$ size.Then perform a convolution operation on the feature data to obtain output data of $[H, W, \mathit{Filters}]$ size. The module can be expressed as Eq. 1, where $y_j^{(i)}$ is the jth type output of the ith sample.$F_{conv}(x, [m, n, \mathit{filters}])$ is a convolution operation on a sample using a convolution kernel of size $[m, n, \mathit{filters}]$.$\mathit{Concat}(x1, x2, \ldots)$ is the operation of

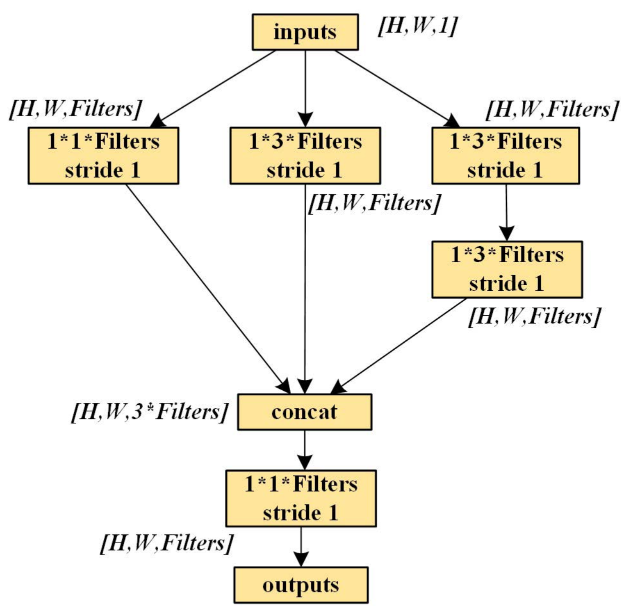


FIGURE 5. Inception block.

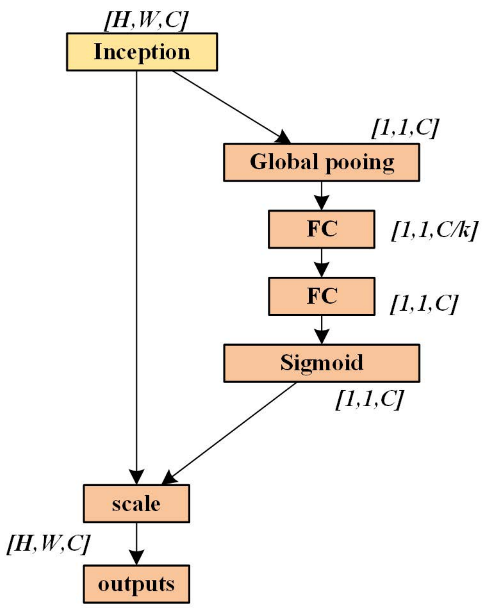


FIGURE 6. SE block.

Algorithm 1 Sequeeze-and-Excitation

Input: Feature data $\mathbf{X}$ and $\mathbf{X} \in \mathbf{R}^{(H\times W\times C)}$
Output: $\mathbf{X}' \in \mathbf{R}^{(H\times W\times C)}$
Initialize: $\mathbf{W}_1 \in \mathbf{R}^{(C\times \frac{C}{k})}$ $\mathbf{W}_2 \in \mathbf{R}^{(\frac{C}{k}\times C)}$
Where k is an integer greater than 0, taken 4 in this paper.
Sequeeze: Calculate $Z \in \mathbf{R}^{(1\times C)}$:
$z_c = \frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W} x_c(i,j)$
Where $x_c(i,j)$ is the feature data of the (i,j) position of the cth channel of the input data and z_c is converted by the cth channel.
Excitation: Calculate $\mathbf{S} \in \mathbf{R}^{(1\times C)}$:
$\mathbf{S} = \delta(\sigma(\mathbf{Z}\mathbf{W}_1)\mathbf{W}_2)$
Where σ is the ReLU function and δ is the Sigmoid function.
Scale: Calculate $\mathbf{X}' \in \mathbf{R}^{(H\times W\times C)}$:
$x_{c'}(i,j) = x_c(i,j) \times z_c$
Where $x_{c'}(i,j)$ is the feature data of the (i,j) position of the cth channel of the output data.
Return: $\mathbf{X}'$

connecting to the input samples. $y_o^{(i)}$ is the output of the ith sample.

$$
\begin{aligned}
y_1^{(i)} &= F_{conv}(x^{(i)}, [1, 1, filters]) \\
y_2^{(i)} &= F_{conv}(x^{(i)}, [1, 3, filters]) \\
y_3^{(i)} &= F_{conv}(F_{conv}(x^{(i)}, [1, 3, filters]), [1, 3, filters]) \\
y_o^{(i)} &= F_{conv}(Concat(y_1^{(i)}, y_2^{(i)}, y_3^{(i)}), [1, 1, filters]) \qquad (1)
\end{aligned}
$$

The Sequeeze-and-Excitation (SE Block) module is another important module in the primary classifier. Fig. 6 shows the structure of the module. SE Block is considered to improve network performance at the feature channel level. The structure can automatically acquire the importance of each feature channel by learning. It can enhance useful features and suppress the features that are less useful for current tasks according to this importance.

First, the input data of the length and width and the number of feature channels $[H, W, C]$ are compressed along the spatial dimension into data of the scale $[1, 1, C]$. This is to turn each two-dimensional feature channel into a real number which has a global receptive field to some extent. And the dimension of the output data matches the number of feature channels of the input data. Then, the weight of each channel is generated by parameters that can be learned, and each weight value is normalized between 0∼1. Finally, the input feature data is weighted layer by layer, and the original features are recalibrated in the channel dimension. The pseudo code for the Sequeeze-and-Excitation module is shown in **Algorithm** 1 [17].

This paper combines Inception Block and SE Block to design the sEMG signal the feature extraction layer of the primary classifier, named IMS Layer which is shown in Fig. 7. First, the data is extracted by Inception Block for multi-scale feature extraction. Then, the data is dimensioned through the Max Pooling layer. Finally, it is recalibrated by the importance of each feature channel of SE Block. In the Inception Block, N is the number of filters used for convolution. And f represents the activation function applied for the output. In Max Pooling, H and W are the length and width of the pooled core, respectively.

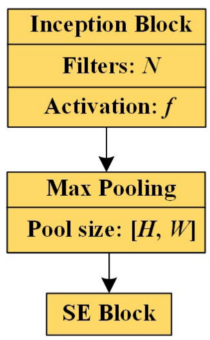


FIGURE 7. IMS layer.

The primary classifier designed consists mainly of the Input Layer, IMS Layer, Fully-connected (FC) Layer and output layer, which is shown in Fig. 8. In this network, the sEMG signal processed can be directly input, and the feature of the sEMG signal can be automatically extracted. The pre-processed sEMG signal will be input to the four-layer IMS Layer for multi-scale feature extraction. Next, the multi-channel feature data is transformed into a one-dimensional feature vector, and the three-layer fully connected layer is applied for movements classification.

The selection of network parameters and activation functions for each layer is given in TABLE 1. It is worth noting that CRelu function is selected as the activation function for

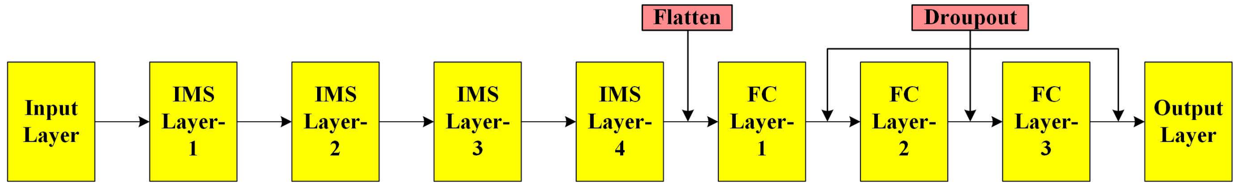


FIGURE 8. Structure of CNN-based primary classifier network.

TABLE 1. Network parameters and activation functions selected in this paper .

Name	Number	Parameters	Activation
IMS Layer	1	Filters:32 Pool size:[1, 2]	CRelu
	2	Filters:32 Pool size:[1, 2]	
	3	Filters:64 Pool size:[1, 2]	
	4	Filters:64 Pool size:[1, 5]	
FC Layer	1	Units:1024	
	2	Units:512	
	3	Units:128	
Output Layer	N/A	Units:40	Softmax

both IMS Layer and FC Layer in the network. CRelu is an improved form of the Relu function. If the output variable is x, the output is activated by the CRelu function is [Relu(x), Relu(-x)]. When Relu is used for activation, the negative information extracted by the network will be erased and result in network redundancy, which can be solved using CRelu [18]. Moreover, when tested with CRelu, the accuracy rate is about 1% higher than that with Relu. Droupout is added between FC layers to prevent overfitting.

D. STACKING BASED SECONDARY CLASSIFIER

Stacking ensemble method, an integrated learning method, combines multiple classification or regression models through a meta-classifier or a meta-regressor. The base layer model is trained on the complete training data set and the outputs will be applied for new model training. **Algorithm** 2 shows the general algorithmic process of Stacking [19].

We use DFT and DWPT to obtain the frequency domain and time-frequency domain data of the sEMG signal.The time domain, frequency domain and time-frequency domain data are used to train the primary classifier to obtain three different classifiers.Stacking method is applied to integrate the results of the three different primary classifiers to improve performance.

Fig. 9 shows the network structure of our secondary classifier based on the Stacking method. The three inputs of the secondary classifier, Outputs-1, Outputs-2, and Outputs-3, are the results of three primary classifiers (trained by data in time domain, frequency domain, and time-frequency domain, respectively). We form a three-channel feature map that is recalibrated by SE Block for the importance of these primary classification results, highlighting important primary classification results and suppressing less important primary classification results. The calibrated feature map is stretched into a one-dimensional feature vector, and the classification result is output through two layers of droupout fully connected layers.

Algorithm 2 Stacking

Input: Training data $D = \{x_i, y_i\}_{i=1}^{m}$
Output: Ensemble classifier H
Step 1: Learn base-level classifiers
 For $t = 1$ to T **do**
  learn h_t based on D
 end for
Step 2: Construct new data set of predictions
 For $i = 1$ to m **do**
  $D_h = \{x_i', y_i\}$ where $x_i' = \{h_1(x_1), ..., h_T(x_i)\}$
 end for
Step 3: Learn a meta-classifier
 Learn H based on D_h
Return: H

E. LOSS AND OPTIMIZATION

The primary classifier and the secondary classifier designed in this paper are all multi-classifiers. Cross Entropy is used as the loss function in the form Eq. 2. Where M is the batch size of the training sample, N is the number of classifications of the output, $y_j^{(i)}$ is the probability that the sample j is actually the ith class, and $\hat{y}_j^{(i)}$ is the probability value of the ith class for the sample j prediction.

$$loss = -\frac{1}{M}\sum_{j=1}^{M}\sum_{i=1}^{N} y_j^{(i)} \log \hat{y}_j^{(i)} \tag{2}$$

The Adam optimization method combines the advantages of both the AdaGrad and RMSProp optimization algorithms. The first moment estimation (mean of the gradient) and the second moment estimation (gradient uncentered variance) are comprehensively considered to calculate update step size [20]. In general, the Adam method is a better optimization method in most cases. Therefore, **Adam** was chosen

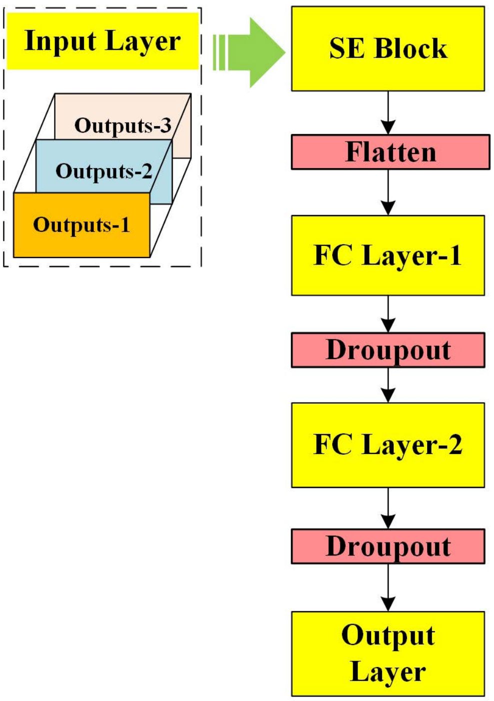


FIGURE 9. Structure of stacking-based secondary classifier network.

as the optimization method for the primary classifier and the secondary classifier.

III. RESULTS

A. DATA SET

NinaPro [21] is an open source project that aims to help EMG prosthetics research through the open sEMG dataset. NinaPro DB5 dataset is applied for the model's train and evaluation in this work. It mainly contains sEMG signal data for 52 different hand movements of 10 subject in DB5 dataset and some examples of hand movements are listed in Fig. 10. In DB5, 52 different types of hand movements are divided into three parts [22], [23]: **Basic movement of the finger (Exercise A)**, **Isometric, isotonic hand configurations and basic wrist movements (Exercise B)**, **Grasping and functional movements (Exercise C)**. Basic information for 10 subjects is listed in TABLE 2, including: Laterality, Gender, Age, Height, and Weight. According to the prompts on the computer screen, each subject finished the designated movement. For each movement, it required repeating 6 times, with 5s for action span as well as 3s for relaxation span in each time.

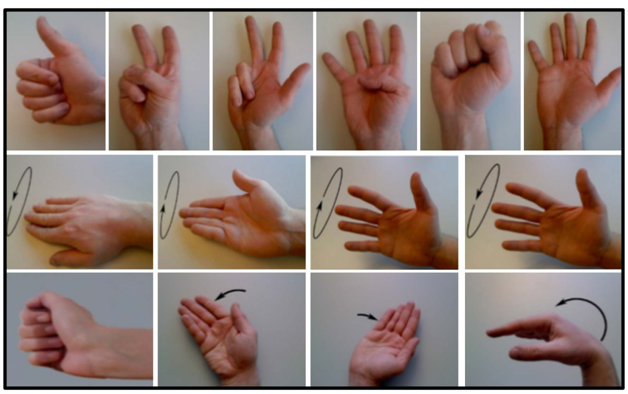


FIGURE 10. Some movements in NinaPro DB5.

TABLE 2. Subject basic information.

Subject	Laterality	Gender	Age	Height	Weight
01	Right Handed	Male	23	187cm	67kg
02	Right Handed	Male	28	187cm	75kg
03	Right Handed	Male	28	170cm	63kg
04	Right Handed	Female	22	156cm	52kg
05	Right Handed	Female	28	160cm	61kg
06	Right Handed	Male	24	170cm	65kg
07	Right Handed	Male	32	172cm	78kg
08	Right Handed	Male	31	170cm	74kg
09	Right Handed	Male	34	176cm	68kg
10	Right Handed	Male	30	173cm	83kg

The acquisition device used in the DB5 dataset is a double MYO bracelet that is worn on the arm of the human body and staggered by 22.5° between the two bracelets. Each MYO bracelet has eight sets of acquisition electrodes that capture the sEMG signal at a 200Hz sampling frequency with a built-in ADC resolution of 8bit. Therefore, the DB5 data set is 16 channels of sEMG signal data, and its value is between -128 and 128.

The MYO bracelet is a wearable sEMG acquisition device. Although the accuracy of the data collection is lower than that of specialized medical acquisition equipment. But its low cost is very suitable for large-scale use. Parallel wear of multiple MYO bracelets makes it easy to achieve multi-channel extended acquisition. We believe that using this device as a human-computer interaction interface to help people with disabilities control prosthetics is of certain research significance. Therefore, we designed a recognition model for the sEMG signals collected by such devices, and selected the DB5 data set to test the performance of the model.

In this paper, the hand movements data in NinaPro DB5 is used for training and testing. Each movements in the data set was repeated 6 times. We used the first 4 replicates as the training set and the last 2 replicates as the test set. Fig. 11. is the result for division of 40 movements data from subject 5, which shows the proportion of each type of data in the current collection. Most of the subsequent experiments were performed using 40 movements in Exercise B and Exercise C in DB5, but a few of them also used Exercise A.

B. MODEL OPTIMIZATION AND FEATURE VISUALIZATION

Firstly, the primary classifier is trained and evaluated using loss and accuracy as the criteria. Fig. 12 shows the training and evaluation of subject 5's sEMG data (40 movements including Exercise B and Exercise C).We use the time domain, frequency domain and time-frequency domain data

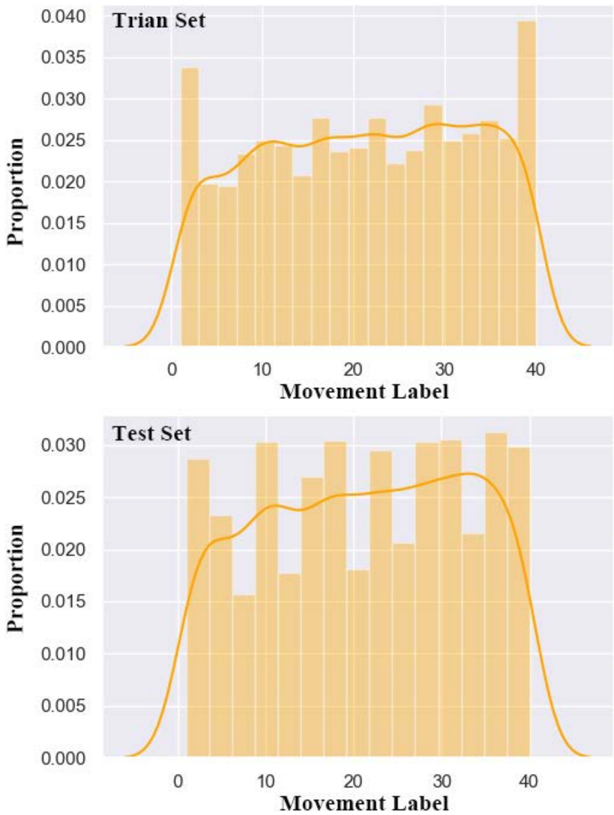


FIGURE 11. Subject 5's sEMG data of different hand movements is divided into training set and testing set.

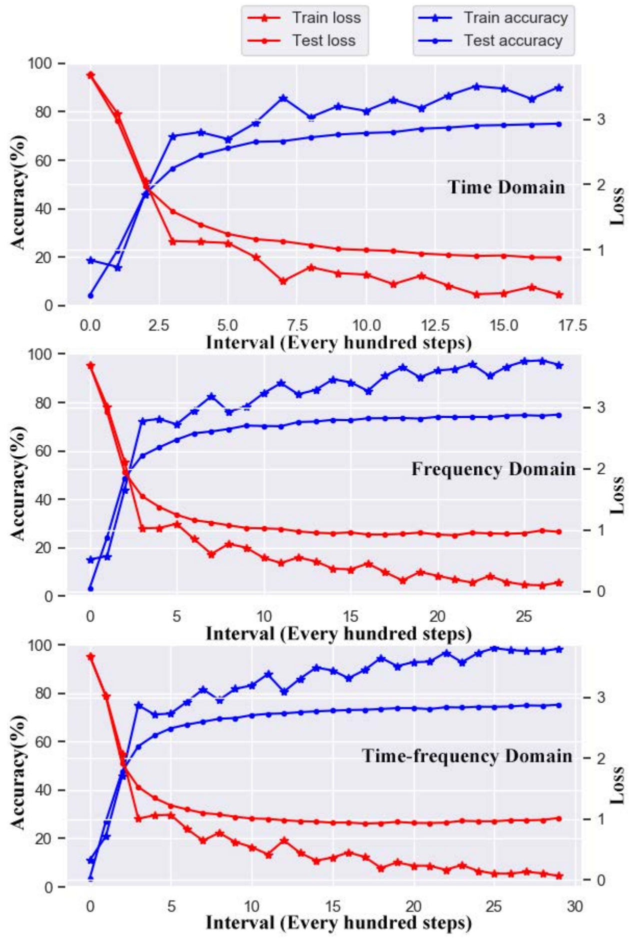


FIGURE 12. Three primary classifiers' accuracy and loss curve which is trianed and evaluated by using time domain, frequency domain and time-frequency, respectively.

of the sEMG signal to train and evaluate three different primary classifiers, respectively. The blue and red curves in the figure are the accuracy and loss curves of three different primary classifiers in training and evaluation.In this paper, the method of batch training (the batch size is 500 samples per time) is used to train the model. The accuracy and loss of the training set and test set are recorded every 100 steps in the training process. The horizontal axis in the figure represents the number of steps of training, and the double vertical axis represents accuracy and loss, respectively. When using the time domain, frequency domain, and time-frequency domain data for primary classifiers training, the accuracy before 500 steps rises sharply and the loss decreases sharply. After 500 steps, the upward trend of accuracy and the downward trend of error will slow down and converge to a better result. The three different classifiers which is trained by time domain, frequency domain and time-frequency domain data respectively can reach test accuracy 76%, 75% and 75%.

Then, the secondary classifier is trained and evaluated. The outputs of the three primary classifiers trained in the time domain, frequency domain, and time-frequency domain are used as features for the training of the secondary classifier. Fig. 13 also shows the accuracy and loss curves for the secondary classifiers trained and evaluated on subject 5 data (40 movements including Exercise B and Exercise C). Before 200 steps, the error dropped sharply and the accuracy

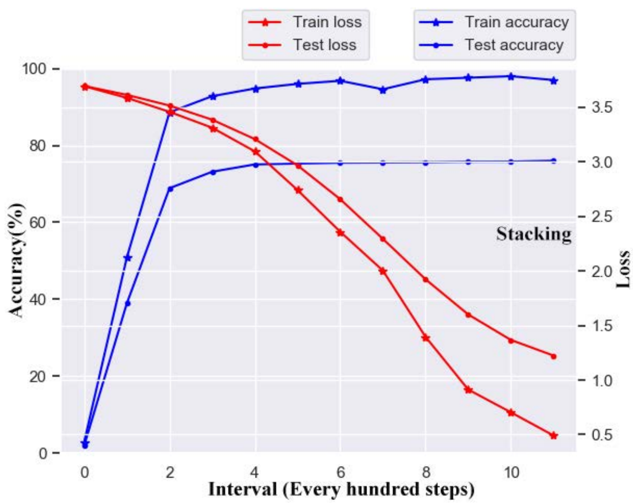


FIGURE 13. Secondary classifier's accuracy and loss curve.

also rose linearly. At approximately 1000 steps of training, the accuracy on the training set and test set is near optimal. Three primary classifiers trained in the data from subject 5 are integrated by secondary classifier can reach 77% accuracy.

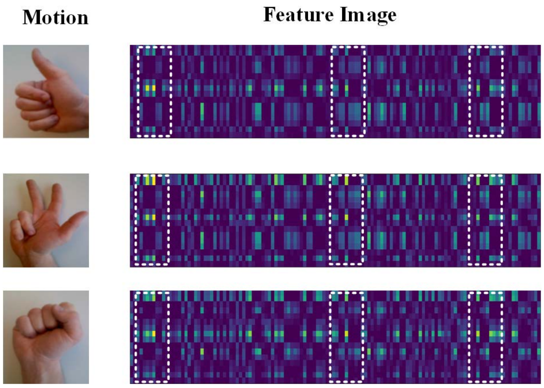


FIGURE 14. Three gesture feature map visualization of subject 5.

We also takes the three actions of subject 5 as an example making visualization of feature extracted by primary classifier in Fig. 14. The feature is generated by the last SE Block of the primary classifier, with a size of 16*1*128. It can represent 128 features of 16-channel sEMG data extracted by the primary classifier. We convert it into a 16*128 two-dimensional matrix and visualize it using images. White frames on the three feature maps show the differences that can be distinguished by the naked eye. But the feature of extracting signal data applying convolution operation is not as easy to understand as the feature extracted from image.

C. COMPARISON OF DIFFERENT WINDOW LENGTHS AND SLIDING LENGTHS

Different sliding lengths and different window lengths are used to perform sliding segmentation on sEMG signal data. TABLE 3, TABLE 4, TABLE 5 are the results of primary classifiers training and testing using data (40 movements of Exercise B and Exercise C) generated by different sliding lengths. TABLE 3 is the accuracy tested with time domain data. TABLE 4 is the accuracy tested with frequency domain data. TABLE 5 is the accuracy tested with time-frequency domain data. We performed the experiments using data from 10 subjects and obtain the mean of the accuracy.

TABLE 3. Results of training on time domain data using different sliding lengths.

Windowing/Sliding	200ms/80ms	200ms/100ms	200ms/125ms
1	70.58%	70.47%	70.15%
2	73.92%	72.92%	71.43%
3	77.58%	76.17%	75.99%
4	65.81%	64.61%	64.08%
5	76.53%	76.16%	74.94%
6	70.45%	70.98%	68.44%
7	62.79%	63.37%	62.41%
8	64.47%	64.03%	62.22%
9	73.51%	73.45%	72.54%
10	74.31%	74.93%	75.90%
avg	71.00%	70.71%	69.81%
std	5.13%	5.01%	5.36%

Fig. 15 shows comparison of the accuracy of different primary classifiers which are trained by data splitting

TABLE 4. Results of training on frequency domain data using different sliding lengths.

Windowing/Sliding	200ms/80ms	200ms/100ms	200ms/125ms
1	68.41%	68.96%	68.69%
2	72.54%	71.04%	71.83%
3	73.49%	73.56%	74.42%
4	63.03%	65.65%	64.53%
5	74.63%	73.78%	74.41%
6	68.12%	69.19%	68.20%
7	63.47%	62.90%	62.24%
8	62.62%	62.76%	62.16%
9	71.07%	72.59%	73.49%
10	74.91%	74.21%	74.16%
avg	69.23%	69.46%	69.41%
std	4.84%	4.39%	5.00%

TABLE 5. Results of training on time-frequency domain data using different sliding lengths.

Windowing/Sliding	200ms/80ms	200ms/100ms	200ms/125ms
1	69.42%	70.41%	69.81%
2	71.63%	71.45%	71.57%
3	74.03%	75.37%	73.78%
4	64.14%	65.69%	63.29%
5	74.50%	74.78%	74.55%
6	69.71%	69.00%	68.64%
7	62.01%	62.07%	61.11%
8	62.34%	62.52%	61.68%
9	72.67%	73.93%	72.16%
10	74.19%	74.11%	74.56%
avg	69.46%	69.93%	69.12%
std	4.92%	5.00%	5.27%

with different sliding lengths. Regardless of using time domain, frequency domain or time-frequency domain data, the performance of models trained utilizing data segmentated with 80ms sliding length or 100ms sliding length is better than 125ms. The performance of the models trained with data splitting using 80ms or 100ms sliding length is almost the same. Therefore, a system with very high latency requirements should use a sliding length of 80 ms or less for sEMG data segmentation. But this has also led to an increase in training data. The following comparative experiments in this paper are all divided by the sliding length of 100ms.

Based on the 100ms sliding length, primary classifier and secondary classifier are trained and evaluated using sEMG data (40 movements of Exercise B and Exercise C) segmented with window lengths of 200ms and 300ms. TABLE 6 and TABLE 7 are the accuracy of models which are trained by using data splitted with window lengths of 200ms and 300ms. We also performed the experiment on the sEMG data of 10 subjects and obtained the average accuracy. Three different primary classifiers trained by the data segmented with 200ms window length can achieve an accuracy of 70.92%±9.95% (time domain), 69.66%±9.67% (frequency domain), and 69.79%±9.80% (time-frequency domain). Accuracy with secondary classifier integration can reach 72.09%±10.11% (stacking). Three different primary classifiers trained by the data segmented with 300ms window length can achieve an accuracy of 75.02%±9.78%

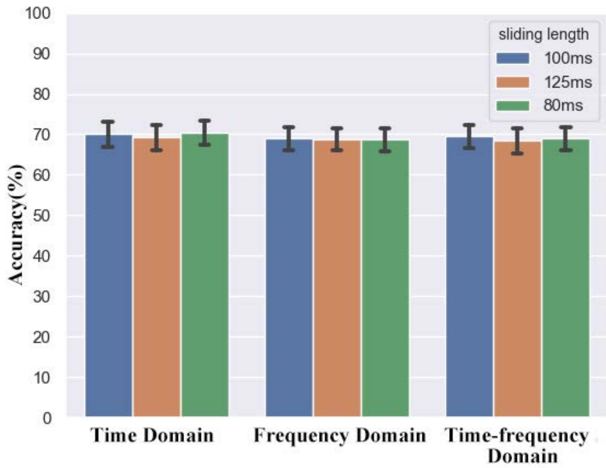


FIGURE 15. Comparison of accuracy with different sliding lengths.

TABLE 6. Results of training using 200ms window length and 100ms sliding length segmentation.

Model	Time	Frequency	Time-frequency	Stacking
1	71.09%	69.23%	69.03%	72.65%
2	73.14%	71.34%	71.38%	73.73%
3	76.62%	74.13%	75.22%	77.94%
4	65.04%	64.02%	64.12%	66.71%
5	76.60%	75.65%	75.58%	77.32%
6	71.05%	69.61%	69.03%	72.07%
7	63.33%	62.10%	62.10%	63.40%
8	64.07%	63.45%	63.86%	65.59%
9	73.04%	72.36%	73.41%	74.87%
10	75.25%	74.71%	74.14%	76.64%
avg	70.92%	69.66%	69.79%	72.09%
std	5.08%	4.94%	5.00%	5.16%

TABLE 7. Results of training using 300ms window length and 100ms sliding length segmentation.

Model	Time	Frequency	Time-frequency	Stacking
1	74.87%	71.93%	74.05%	75.71%
2	77.10%	74.68%	76.01%	78.16%
3	80.61%	77.88%	78.64%	81.01%
4	71.15%	69.73%	69.92%	72.49%
5	80.30%	78.34%	80.06%	81.10%
6	75.51%	72.62%	73.28%	75.83%
7	66.16%	65.65%	67.12%	67.77%
8	68.58%	67.27%	67.23%	69.60%
9	75.88%	74.60%	76.66%	77.70%
10	80.00%	78.17%	78.35%	80.81%
avg	75.02%	73.09%	74.13%	76.02%
std	4.99%	4.51%	4.70%	4.75%

(time domain), 73.09%±8.84% (frequency domain), and 74.13%±9.20% (time-frequency domain). Accuracy with secondary classifier integration can reach 76.02%±9.31% (stacking). The above results were calculated with a confidence level of 95%. The model trained by the data segmentated with 300ms window length is more accurate than segmentated with 200ms window length. As the number of data points increases, the amount of useful information that can be extracted increases, and the corresponding delay is also longer. Therefore, comprehensive consideration is needed in practical applications.

D. COMPARISON OF DIFFERENT MODELS

Experiments were carried out applying the Exercise A and Exercise B action groups of the DB5 database in [13]. It describes the accuracy of hand movements classification using the LDA model, SVM model and RMS, TD [24], HIST and mDWT. The author also designed the LCNN model for the classification of hand movements and conducted experiments to determine the accuracy of the classification.We also used the same experimental method as the one in [13] for the model designed in this paper. Fig. 16 is the comparison on accuracy between the primary classifier, secondary classifier designed by us with the model in [13]. Three primary classifiers (which are trained by time domain, frequency domain and time-frequency data) or secondary classifier performed better than LDA, SVM, and LCNN. In the case of training model applying Exercise A sEMG data, our best primary classifiers improved 5% over LCNN in accuracy. And secondary classifiers have an 7% improvement over LCNN. In the case of Exercise B sEMG data, our best primary classifiers improved 13% over LCNN in accuracy. And secondary classifiers have an 14% improvement over LCNN.

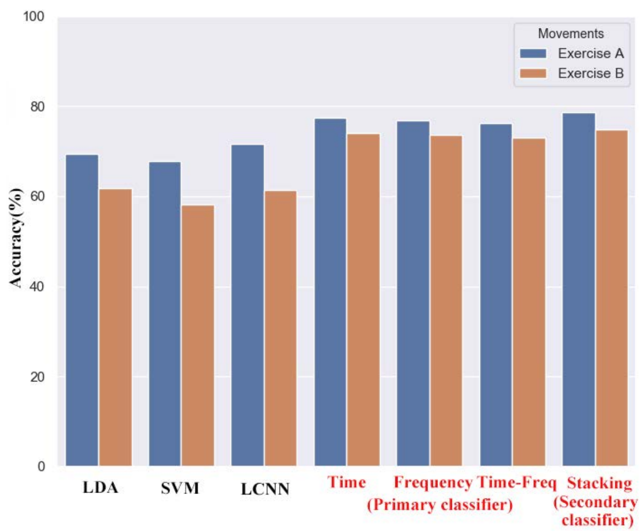


FIGURE 16. Comparison of training results on Exercise A and Exercise B.

The sEMG signal data of 40 hand movements is segmented using 200ms window length and 100ms sliding length in [7]. Four features of RMS, TD, HIST and mDWT were extracted, and SVM and Random Forests models were used for classification training. In this paper we use the same method to segment the sEMG data of 40 movements from Exercise B and Exercise C. And we apply the data processed at same to train our models. Fig. 17 is a comparison between different model classification accuracy. The best accuracy of SVM or Random Forests which is trained on the data of DB5-1 (8-channel sEMG data collected on the MYO bracelet on the upper side of the arm) is 55.31%. And the best accuracy in

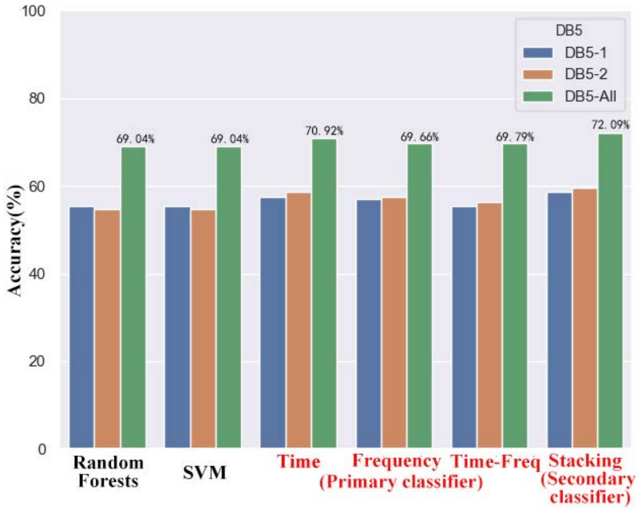


FIGURE 17. Comparison of training results on segmenting the 40 movements data of Exercise B and Exercise C using the 200ms window length and 100ms sliding length.

three different primary classifiers which are trained by time domain, frequency domain and time-frequency domain data is 57.54%. The accuracy of secondary classifier increased to 58.74%. In the case of the DB5-2 (8-channel sEMG data collected from the MYO bracelet under the arm), the best accuracy of SVM or Random Forests was 54.76%. And the best accuracy in three different primary classifiers is 58.73%. The accuracy of secondary classifier increased to 59.57%. In the case of the DB5-All (16-channel sEMG data collected from double MYO bracelet), the best accuracy of SVM or Random Forests was 69.04%. And the best accuracy in three different primary classifiers is 70.92%. The accuracy of secondary classifier increased to 72.09%. As a result, the accuracy of the primary classifier and the secondary classifier designed by us is better than SVM and Random Forests. Comparing to SVM and Random Forests, the accuracy of the secondary classifier we desgin increased by 3.43%, 4.74%, and 3.05% on DB5-1, DB5-2, and DB5-All.

IV. CONCLUSION

The combination of CNN and Stacking ensemble learning is used for sEMG hand movements classification, which can overcome the limitation of the feature engineering requiring better feature quality. Three different primary classifiers based on CNN are trained using time domain, frequency domain and time-frequency domain data of the sEMG signal. The primary classifier is mainly composed of Inception Block, SE Block and fully-connected layer, in order to directly extract and classify the sEMG signals after preprocessing. Based on the stacking ensemble learning algorithm, a secondary classifier is designed to integrate the three primary classifiers to further improve the accuracy of sEMG hand movements classification. Secondary classifier is trained using the feature data which is generated from three primary classifiers. In the experiments, the primary classifier and the secondary classifier were trained and tested using the subject 5's sEMG data as an example firstly. Next, we perform the experiments on the data segmented by different sliding lengths and different window lengths. Experiments were performed on sEMG data of 10 subjects, respectively, and the mean of the accuracy was obtained. Finally, the models designed are trained and tested using the experimental methods in [7] and [13]. The experimental results demonstrate the primary classifier and secondary classifier designed have a better peformance compared with LDA, SVM, LCNN and Random Forests.

ACKNOWLEDGMENT

The authors are very grateful to the open source dataset provided by the NinaPro project team, which is an important prerequisite for the successful implementation of this article. Furthermore, they would like to thank NVIDIA for supporting to this project. All of the experiments in this article worked smoothly on NVIDIA TITAN XP. *(Shu Shen and Kang Gu are co-first authors.)*

REFERENCES

[1] R. Alam, S. R. Rhivu, and M. A. Haque, "Improved gesture recognition using deep neural networks on sEMG," in *Proc. Int. Conf. Eng., Appl. Sciences, Technol. (ICEAST)*, Jul. 2018, pp. 1–4.

[2] S. Zhou, K. Yin, Z. Liu, F. Fei, and J. Guo, "sEMG-based hand motion recognition by means of multi-class adaboost algorithm," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, Dec. 2018, pp. 1056–1061.

[3] R. Meattini, S. Benatti, U. Scarcia, D. De Gregorio, L. Benini, and C. Melchiorri, "An sEMG-based human-robot interface for robotic hands using machine learning and synergies," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 8, no. 7, pp. 1149–1158, Jul. 2018.

[4] N. Jose, R. Raj, P. K. Adithya, and K. Sivanadan, "Classification of forearm movements from semg time domain features using machine learning algorithms," in *Proc. TENCON IEEE Region Conf.*, Nov. 2017, pp. 1624–1628.

[5] I. Kuzborskij, A. Gijsberts, and B. Caputo, "On the challenge of classifying 52 hand movements from surface electromyography," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Aug. 2012, pp. 4931–4937.

[6] A. Gijsberts, M. Atzori, C. Castellini, H. Müller, and B. Caputo, "Movement error rate for evaluation of machine learning methods for sEMG-based hand movement classification," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 22, no. 4, pp. 735–744, Jul. 2014.

[7] S. Pizzolato, L. Tagliapietra, M. Cognolato, M. Reggiani, H. Müller, and M. Atzori, "Comparison of six electromyography acquisition setups on hand movement classification tasks," *PloS ONE*, vol. 12, no. 10, Oct. 2017, Art. no. e0186132.

[8] K. Xing, Z. Ding, S. Jiang, X. Ma, K. Yang, C. Yang, X. Li, and F. Jiang, "Hand gesture recognition based on deep learning method," in *Proc. IEEE 3rd Int. Conf. Data Sci. Cyberspace (DSC)*, Jun. 2018, pp. 542–546.

[9] M. Atzori, M. Cognolato, and H. Müller, "Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands," *Frontiers Neurorobotics*, vol. 10, p. 9, Sep. 2016.

[10] X. Zhai, B. Jelfs, R. H. M. Chan, and C. Tin, "Self-recalibrating surface EMG pattern recognition for neuroprosthesis control based on convolutional neural network," *Frontiers Neurosci.*, vol. 11, p. 379, Jul. 2017.

[11] Y. He, O. Fukuda, N. Bu, H. Okumura, and N. Yamaguchi, "Surface emg pattern recognition using long short-term memory combined with multilayer perceptron," in *Proc. 40th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2018, pp. 5636–5639.

[12] Y. Hu, Y. Wong, W. Wei, Y. Du, M. Kankanhalli, and W. Geng, "A novel attention-based hybrid CNN-RNN architecture for sEMG-based gesture recognition," *PloS ONE*, vol. 13, no. 10, Oct. 2018, Art. no. e0206049.

[13] Y. Wu, B. Zheng, and Y. Zhao, "Dynamic gesture recognition based on LSTM-CNN," in *Proc. Chin. Autom. Congr. (CAC)*, Nov. 2018, pp. 2446–2450.

[14] B. Hudgins, P. Parker, and R. N. Scott, "A new strategy for multifunction myoelectric control," *IEEE Trans. Biomed. Eng.*, vol. 40, no. 1, pp. 82–94, Jan. 1993.

[15] M. Simão, N. Mendes, O. Gibaru, and P. Neto, "A review on electromyography decoding and pattern recognition for human-machine interaction," *IEEE Access*, vol. 7, pp. 39564–39582, 2019.

[16] P. Virdi, Y. Narayan, P. Kumari, and L. Mathew, "Discrete wavelet packet based elbow movement classification using fine Gaussian SVM," in *Proc. IEEE 1st Int. Conf. Power Electron., Intell. Control Energy Syst. (ICPEICES)*, Jul. 2017, pp. 1–5.

[17] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[18] W. Shang, K. Sohn, D. Almeida, and H. Lee, "Understanding and improving convolutional neural networks via concatenated rectified linear units," 2016, *arXiv:1603.05201*. [Online]. Available: https://arxiv.org/abs/1603.05201

[19] L. Breiman, "Stacked regressions," *Mach. Learn.*, vol. 24, no. 1, pp. 49–64, 1996.

[20] D. P. Kingma and J. A. Ba, "Adam: A method for stochastic optimization," 2019, *arXiv:1412.6980*. [Online]. Available: https://arxiv.org/abs/1412.6980

[21] N. Team, *Ninaweb*. Accessed: Feb. 25, 2019. [Online]. Available: http://ninapro.hevs.ch/

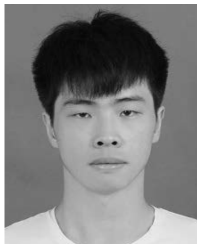
[22] M. Atzori, A. Gijsberts, I. Kuzborskij, S. Elsig, A.-G. M. Hager, O. Deriaz, C. Castellini, H. Müller, and B. Caputo, "Characterization of a benchmark database for myoelectric movement classification," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 23, no. 1, pp. 73–83, Jan. 2015.

[23] M. Anti, A. Gijsberts, C. Castellini, B. Caputo, A.-G. M. Hager, S. Elsig, G. Giatsidis, F. Bassetto, and H. Müller, "Electromyography data for non-invasive naturally-controlled robotic hand prostheses," *Nature*, vol. 1, Dec. 2014, Art. no. 140053.

[24] K. Englehart and B. Hudgins, "A robust, real-time control scheme for multifunction myoelectric control," *IEEE Trans. Biomed. Eng.*, vol. 50, no. 7, pp. 848–854, Jul. 2003.

SHU SHEN received the B.S. degree in information engineering and the M.S. degree in communication engineering from Southeast University, Nanjing, China, in 2005 and 2008, respectively, and the Ph.D. degree in electronic device from the Politecnico di Torino, Turin, Italy, in 2012.

He joined the School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing, China, as an Assistant Professor, in 2013, where he is currently an Associate Professor. He has been a Postdoctoral Researcher of ICT and IoT studies with the Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks, Nanjing, China, since 2015. His current research interests include wireless sensor network, real-time embedded system design, wearable computing, and Internet of Things. He is a member of ACM and CCF.



KANG GU received the B.S. degree in information engineering from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 2018, where he is currently pursuing the M.S. degree in computer technology with the School of Computer Science. His current research interests include deep learning and wearable computing.

XIN-RONG CHEN received the B.S. degree in electronic and information engineering from the Nanjing University of Science and Technology, in 2004, and the M.S. degree in signal and information processing from Southeast University, Nanjing, China, in 2007, and the Ph.D. degree in biomedical engineering from Fudan University, China, in 2014. He was with the Institute of Neuroscience and Medicine -4, Forschungszentrum JÍźlich, Germany, as a Postdoctoral Researcher, from 2016 to 2018. He is currently a Professor with the Academy for Engineering and Technology, Fudan University, China. His current research interests include medical image analysis, computer vision, and image-guided surgery.

MING YANG received the B.S. degree from the Qingdao University of Science and Technology, Qingdao, China. He is currently pursuing the M.S. degree in software engineering with the School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing, China. His current research interest includes human-computer interaction.

RU-CHUAN WANG is currently a Full Professor and a Ph.D. Supervisor with the School of Computer Science and School of Software, Nanjing University of Posts and Telecommunications, Nanjing, China. His current research interests include computer network, cloud computing, big data, and wireless sensor network.

• • •