Motion Classification Based on sEMG Signals Using Deep Learning

Shu Shen[1,2], Kang Gu[1], Xinrong Chen[3,4(✉)], and Ruchuan Wang[1,2]

[1] School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China
{shens,1218043119,wangrc}@njupt.edu.cn

[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
chenxinrong@fudan.edu.cn

[4] Shanghai Key Laboratory of Medical Image Computing and Computer Assisted Intervention, Shanghai 200032, China

Abstract. Nowadays, surface electromyography (sEMG) signal plays an important role in helping physically disabled people during daily life. The development of electronic information technology has also led to the emergence of low-cost, multi-channel, wearable sEMG signal acquisition devices. Therefore, this paper proposes a new motion recognition model based on deep learning to improve the accuracy of motion recognition of sEMG signals. The model uses architecture including 6 convolutional layers and 6 pooling layers which enables the Batch Normalization layer to enhance network performance and prevent overfitting. In the experiment, NinaPro DB5 data set was used for training and testing. The data set consists of sEMG signal data collected by the double Myo armbands which contains data of 52 movements for 10 subjects. The results show that the accuracy of about 90% can be achieved when using 52 sEMG signal data from every single subject or all subjects.

Keywords: Surface electromyogram signal · Deep learning · Wearable device · Motion recognition

1 Introduction

Human muscle is made up of many motor units, and as the muscles move, certain motion units are activated to produce bioelectrical signals. The superposition of bioelectrical signals produces myoelectric signal [1], which is reflected by sEMG signal. sEMG signal records the muscle activity from the skin surface and reflect the generation of myoelectric signals. Moreover, sEMG is a non-invasive method for collecting bioelectrical feedback signal. Therefore, sEMG signal is the critical technology used in motion recognition, which has been widely used in muscle

X. B. Zhai et al. (Eds.): MLICOM 2019, LNICST 294, pp. 563–572, 2019.
https://doi.org/10.1007/978-3-030-32388-2_48

feedback devices such as prosthetic control [2]. How to perform motion recognition based on the sEMG signal can be considered as a problem of pattern recognition. The motion recognition based on the sEMG signal is faced with several challenges and one of the most critical challenges is the original electromyography signals in each channel are non-stationary, non-linear, random and unpredictable. To solve the problem, deep learning methods for classification are proposed and have better accuracy and effectiveness [1].

Deep learning is a new technology in the area of machine learning. Its motivation is derived from the simulation and establishment of the neural network in human brain analysis and learning. Now, deep learning is one of the most popular techniques in image recognition, natural language processing, and speech recognition. At the same time, deep learning is a key technology used for data analysis, processing, and identification by various research teams. In this paper, a motion recognition model based on deep learning is proposed. The motion classification is carried out by the feature extraction of sEMG signals using multi-layer convolutional neural network. The model takes into account the valid information in the sEMG signal, which is usually ignored, and the performance of motion recognition is improved.

The rest of this paper is organized as follows. Section 2 discusses related research on this topic. Section 3 introduces the sEMG signal data set used in this experiment and the method of preprocessing the data. A motion recognition model using deep learning is proposed and the structure of the model is described in detail. The training and testing of the motion recognition model in Sect. 4 describes the experiment results. Section 5 summarizes the full text.

2 Related Work

Due to the development of non-invasive sEMG detection and machine learning, many researchers are committed to identifying human motion through sEMG signals. Many researchers have developed some methods for sEMG signal recognition based on time domain, frequency domain and time-frequency domain. At the same time, many researchers apply machine learning to sEMG recognition.

Pizzolato et al. [3] compared the six sEMG acquisition methods based on the NinaPro project and used SVM and Random Forests method to classify the characteristics of sEMG with a correct rate of 69%. Thakur et al. [4] presented a method which based on wavelet transform and support vector machine to classify the movements of human elbows. There are also some methods using feature extraction and machine learning which achieved good results [5–11]. Cote-Allard et al. [12] proposed a method with migration learning using convolutional neural networks to classify sEMG signals reducing the size of the training set required for model training. A parallel architecture deep convolutional network with five convolutional layers was proposed in [13], which realized the recognition of 17 kinds of actions, and the correct rate reached 82%. Zhou et al. [14] presented the method combining AdaBoost algorithm and BP neural network to identify 8 wrist movements and 12 finger movements. The sEMG signal was proposed to be imaged in [15] and identified using a deep learning method.

3 Motion Recognition

3.1 Frame

This study focused on sEMG signal data acquired by wearable armbands. Such devices are often portable and low power consumption. Although the accuracy is not very high, the cost is much lower than that of professional acquisition equipments. They are very suitable for people with physical disabilities to control prostheses. Figure 1 shows the frame of motion recognition using armbands, including acquisition of **sEMG signals**, **data preprocessing**, **signal feature extraction** and **motion classification**.

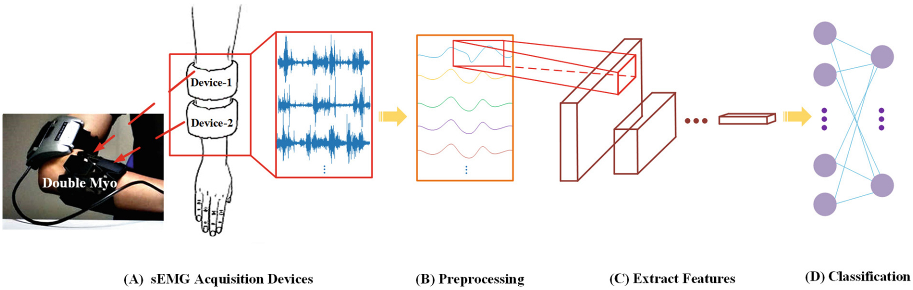

Fig. 1. Motion recognition frame

3.2 Data Set

NinaPro [16] is a project in progress to help research myoelectric prostheses in the hands with open data sets. It now contains seven databases related to the sEMG signal. The data set used in this paper is NinaPro DB5.

Acquisition Protocol. NinaPro DB5 contains a total of 52 motion data, all selected from manual taxonomy and manual robot literature. These actions are divided into three groups, named **Basic movement of the finger, Isometric, isotonic hand configurations and basic wrist movements, Grasping and functional movements.** A schematic representation of the actions from the NinaPro DB5 data set is shown in Fig. 2. During the acquisition of the sEMG signal, the subject is required to repeat the action. Each exercise is repeated for 5 s, then the rest is for 3 s. The NinaPro DB5 database contains a total of 6 repetitions of 52 different exercises (plus rest) performed by 10 intact subjects.

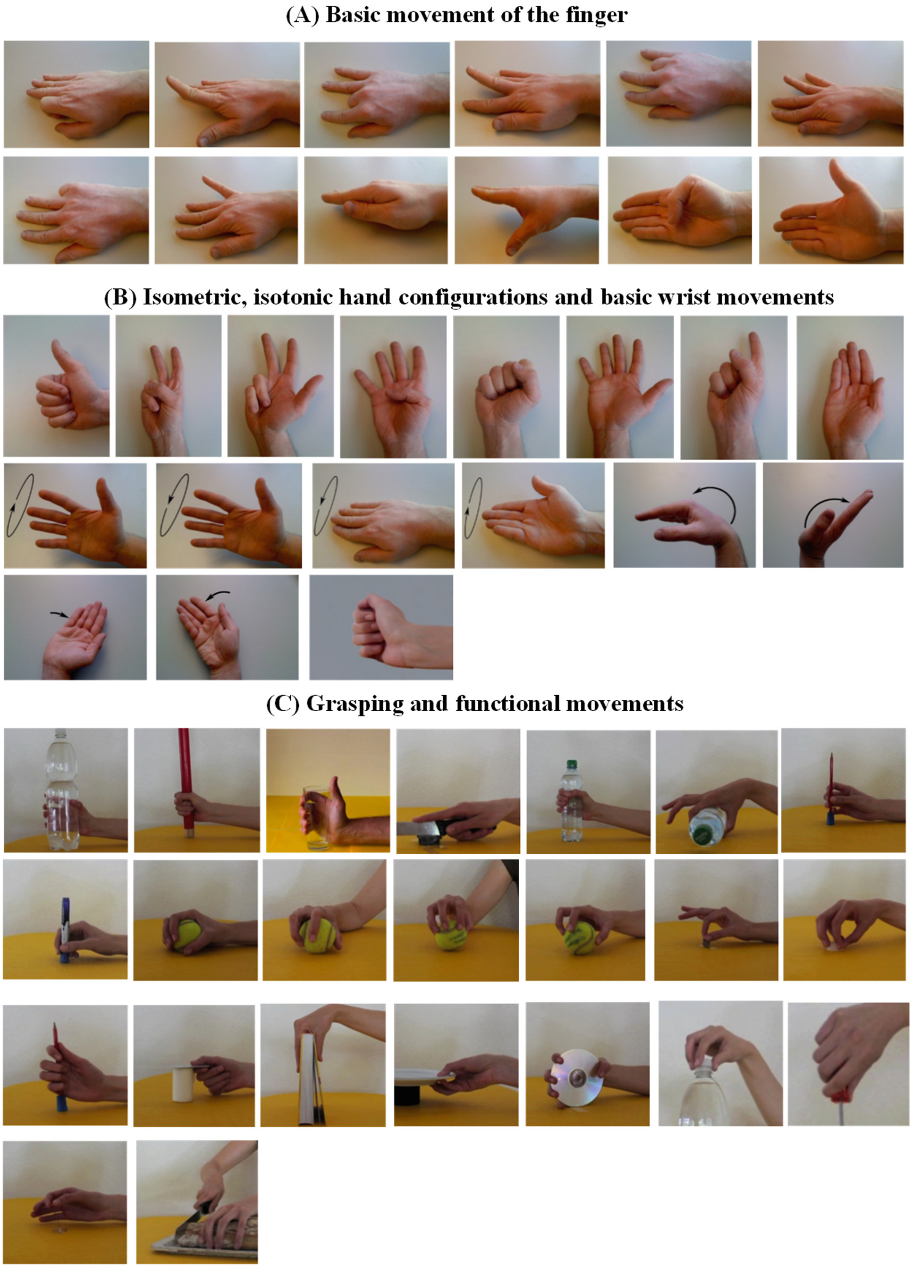

Fig. 2. The 52 movements in DB5

Acquisition Setup. The database has sEMG data for 10 subjects. Part (A) in Fig. 1 shows the acquisition device. Each subject wears two Myo armbands, one after the other, including 16 active single differential wireless electrodes while acquiring the sEMG signal. The top Myo arm strap is placed close to the elbow and the first sensor is placed on the tibial joint. The second Myo arm band is placed behind the first one, close to the hand, tilted 22.5° to provide an extended uniform muscle mapping. Each Myo armband samples the sEMG signal at the rate of 200 Hz.

Preprocessing. The data needs to be pre-processed before training. Because the data used in this paper is sampled at the frequency of 200 Hz and the sampling time for each action is 5 s, the data length of each channel is aligned to 1000. Firstly, data segmentation is performed according to the data labels in the DB5 database, and the sEMG signal data of each action is separated. Then, each channel has a data length greater than 1000 and is intermittently sampled, and less than 1000 is subjected to cubic interpolation. After the length normalization is performed, the amplitude of the data is normalized between 0 and 1. We use Eq. (1) to perform amplitude normalization.

$$A_n = \frac{A_r + 256.0}{512.0} \tag{1}$$

A_r is the original data of the sEMG signal, and A_n is the data after the amplitude is normalized.

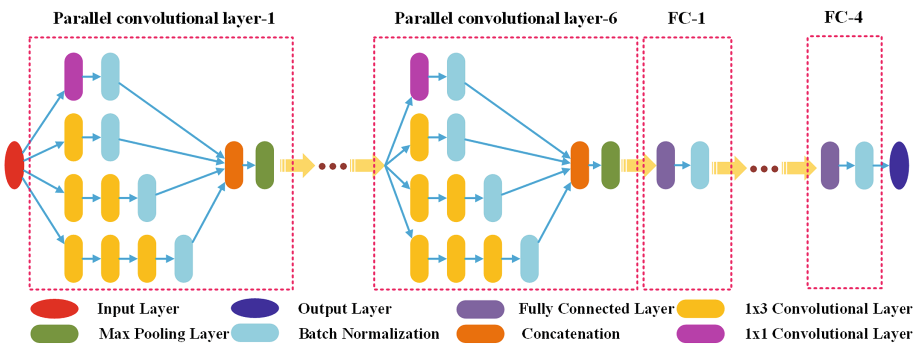

Fig. 3. Model

3.3 Recognition Model

The sEMG signal processed in this paper has 16 channel and each channel has 1000 data. Therefore, in order to complete the recognition of motion, we represent them by a matrix of 16 × 1000 and use the convolutional neural network to extract the features of the sEMG signal in order to complete the recognition of motion. The sEMG motion recognition model based on convolutional neural network proposed for us in Fig. 3. The input layer is 16 × 1000 sEMG signal data. Six parallel convolutional layer stacking structures are used to extract the features of the sEMG signal step by step. The high-dimensional features are then mapped to low dimensions using a 4-layer fully-connected network, which is then classified by the output layer. The parallel convolutional layer uses four types of individual convolutions for feature extracting. At the same time, batch standardization was added after each type of convolution. 1 × 1 and 1 × 3 size one-dimensional convolution kernels are used for feature extraction of single-channel data which reduces interference between channels and improves the performance of the model. The output of each layer of parallel convolutional layers is the Eq. (2).

$$\begin{aligned} y_1^{(i)} &= \mathrm{F_{BN}}(\tanh(\mathrm{F_{conv}}(\mathrm{x^{(i)}}, [1, 1]))) \\ y_2^{(i)} &= \mathrm{F_{BN}}(\tanh(\mathrm{F_{conv}}(\mathrm{x^{(i)}}, [1, 3]))) \\ y_3^{(i)} &= \mathrm{F_{BN}}(\tanh(\mathrm{F_{conv}}(\mathrm{F_{conv}}(\mathrm{x^{(i)}}, [1, 3]), [1, 3]))) \\ y_4^{(i)} &= \mathrm{F_{BN}}(\tanh(\mathrm{F_{conv}}(\mathrm{F_{conv}}(\mathrm{F_{conv}}(\mathrm{x^{(i)}}, [1, 3]), [1, 3]), [1, 3]))) \\ y_o^{(i)} &= \mathrm{Pool}(\mathrm{concat}(\mathrm{y_1^{(i)}}, \mathrm{y_2^{(i)}}, \mathrm{y_3^{(i)}}, \mathrm{y_4^{(i)}}), \mathit{pool\_size}) \end{aligned} \tag{2}$$

Where $y_j^{(i)}, j = \{1, 2, 3, 4\}$ is the jth type output of the sample i and $\mathrm{F_{conv}}(x, [m, n])$ is the convolution operation of the sample x using the convolution kernel of size $[m, n]$. Tanh is the activation function used by the convolutional layer. $\mathrm{F_{BN}}(x)$ is a batch normalization operation on sample x. $\mathrm{concat}(x_1, x_2, \ldots)$ is the operation of connecting to the input samples. $\mathrm{Pool}(x, \mathit{pool\_size})$ is a pooling operation for pool_size size for sample x. $y_o^{(i)}$ is the output of parallel convolutional layer.

The full connection layer adopts the structure of a general multi-layer perceptron. Cross-entropy defined in Eq. (3) is used as the loss function of classification, Where $y^{(i)}$ represents the probability value that is actually class i, and $\hat{y}^{(i)}$ indicates that the prediction is a probability value of class i.

$$Loss = -\sum_i y^{(i)} \log \hat{y}^{(i)} \tag{3}$$

Table 1 shows specific parameters of each layer about convolutional neural network designed proposed. The number of convolution kernel channels in the same layer is constant. We expound the number of specific channels of the convolution kernel in rows 2–7 of the table and the pooling size of each max pooling layer in row 8–13. The number of neurons about fully connected and output layers is in the remaining rows. Convolutional layer and the fully connected layer use tanh as the activation function. The output layer uses softmax as the activation function.

Table 1. Parameters of each layer about convolutional neural network proposed.

Name	Number	Parameter		Activation
Parallel convolutional layer	1	Each convolution kernel channel	4	tanh
	2		8	
	3		16	
	4		32	
	5		64	
	6		256	
Max Pooling layer	1	Pool size	[1, 5]	None
	2		[1, 5]	
	3		[1, 2]	
	4		[1, 2]	
	5		[1, 2]	
	6		[1, 5]	
FC layer	1	Units	2048	tanh
	2		1024	
	3		512	
	4		128	
Output layer	1		52	Softmax

Table 2. Test accuracy of the model proposed in the paper.

Subject number	Train set/Test set accuracy			Average accuracy
	(1, 2)(3, 4)/(5, 6)	(1, 2)(5, 6)/(3, 4)	(3, 4)(5, 6)/(1, 2)	
1	90.38%	93.27%	81.73%	88.64%
2	94.23%	95.19%	84.62%	91.35%
3	97.12%	99.04%	90.38%	95.51%
4	89.42%	93.27%	84.62%	89.10%
5	95.19%	95.19%	81.65%	90.68%
6	90.38%	97.12%	82.69%	90.06%
7	89.40%	93.27%	76.96%	86.54%
8	87.50%	92.31%	77.88%	85.90%
9	93.27%	99.04%	87.51%	93.27%
10	89.42%	93.26%	82.69%	88.46%
All subject	91.06%	93.26%	83.26%	89.45%

4 Simulation

Data of 10 subjects in NinaPro DB5 database is used to train and test the model designed in this paper. The data of 6 repetitions of each action in the DB5 database is divided into three groups of (1, 2), (3, 4), and (5, 6). Each test selects one of them as the test set and the rest as the training set. The average correct rate of three times is used as the final correct rate. The result of the tests on the data of each subject and all subjects is presented in Table 2. The test was confirmed to be 89.95% ± 2.78% on the data of each subject. The accuracy of testing on all subjects was 89.45%.

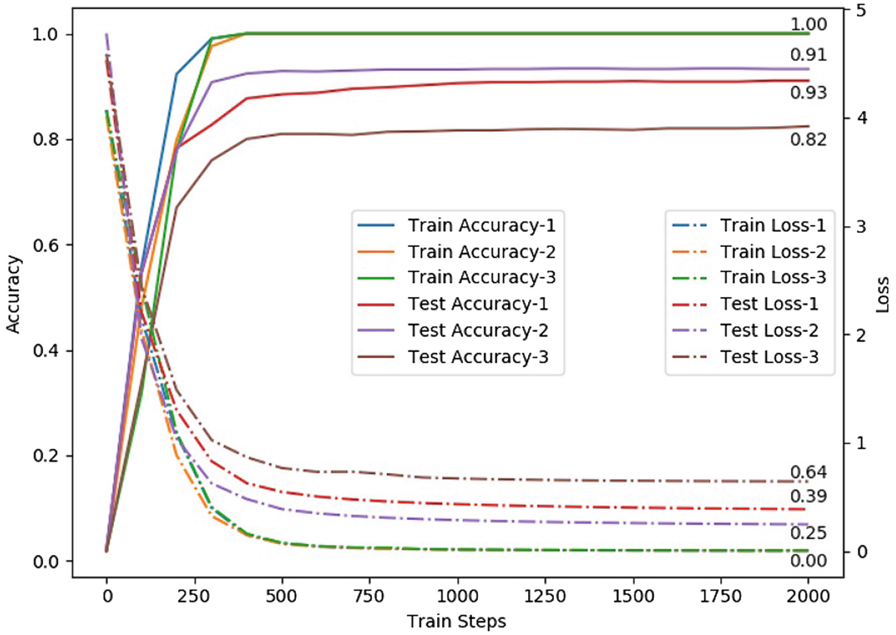

Fig. 4. Accuracy and loss in all subject

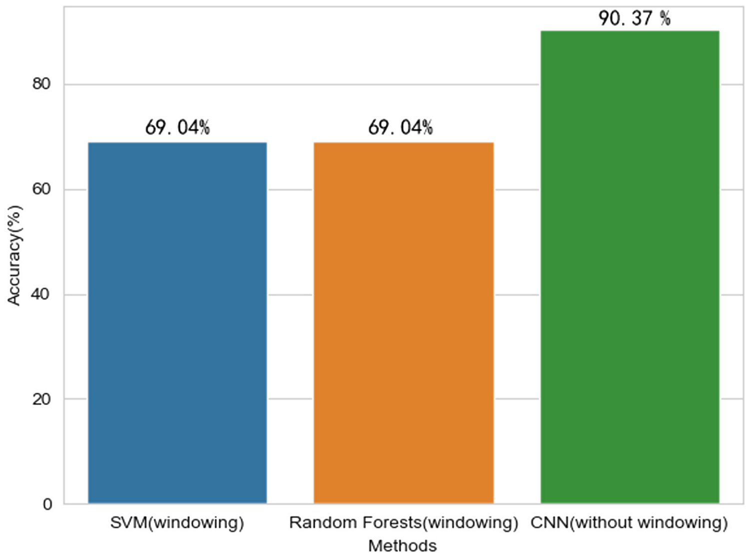

Fig. 5. Comparison

Figure 4 shows a plot of accuracy, error, and number of training steps was tested on the data of all subjects. We conducted a total of 2000 steps on the network and recorded the error and accuracy of the network on the training set and test set for each 100 steps. We also used the data of 41 movements in group B and group C to test the model and compare with the test accuracy of using SVM and Random Forests in [3]. Figure 5 shows the results of the comparison. However, windowing was performed when using SVM and Random Forests for sEMG signal recognition, and the work we did was not windowed. We will study the windowed signals in the next work.

5 Conclusion

This paper proposes a motion recognition model based on sEMG using deep learning. The model uses a multi-layer parallel convolutional neural network to extract features from sEMG signals. Compared with the traditional feature extraction method, it can extract more effective information and improve the accuracy of motion recognition. The NinaPro DB5 data set is used to train in the simulation section and the model is tested on 52 motion data from 10 subjects. The experiment results demonstrate the proposed method has a better performance than both the traditional SVM method and the Random Forests methods. In the future, because windowing has an effect on the performance of the recognition method, we will study the windowed signals in the next work.

Lab Environment. The CPU for the desktop used in the experiment was Intel(R) Core(TM) i7-7800X CPU @ 3.5G Hz (3504 MHz), and the GPU was NVIDIA TITAN Xp (12 GB). The software environment used in the experiment was Tensorflow-gpu 1.12.

Acknowledgment. The authors would like to thank the financial support by the National Science Foundation of P.R. China under Grant Nos. 61401221, 61873131, 61872196, 61701168, 61572261, 61572260, China Postdoctoral Science Foundation under Grant No. 2016M601860, Postgraduate Research & Practice Innovation Program of Jiangsu Province under Grant Nos. SJCX19_0240, SJKY19_0823, NJUPT Teaching Reform Project under Grant No. JG00417JX74.

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 paper. Furthermore, we would like to thank NVIDIA for supporting to this project. All of the experiments in this article worked smoothly on NVIDIA TITAN Xp.

References

1. Alam, R.U., Rhivu, S.R., Haque, M.: Improved gesture recognition using deep neural networks on sEMG. In: 2018 International Conference on Engineering, Applied Sciences, and Technology (ICEAST), pp. 1–4. IEEE (2018)

2. Sayin, F.S., Ozen, S., Baspinar, U.: Hand gesture recognition by using sEMG signals for human machine interaction applications. In: 2018 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA), pp. 27–30. IEEE (2018)
3. Pizzolato, S., Tagliapietra, L., Cognolato, M., Reggiani, M., Müller, H., Atzori, M.: Comparison of six electromyography acquisition setups on hand movement classification tasks. PLoS ONE **12**(10), e0186132 (2017)
4. Thakur, N., Mathew, L.: sEMG signal classification using ensemble learning classification approach and DWT. In: 2018 International Conference on Current Trends towards Converging Technologies (ICCTCT), pp. 1–4. IEEE (2018)
5. Atzori, M., et al.: Electromyography data for non-invasive naturally-controlled robotic hand prostheses. Sci. Data **1**, 140053 (2014)
6. Atzori, M., et al.: Characterization of a benchmark database for myoelectric movement classification. IEEE Trans. Neural Syst. Rehabil. Eng. **23**(1), 73–83 (2015)
7. Xue, Y., Ju, Z., Xiang, K.: Classification of dynamic in-hand manipulation based on SEMG and Kinect. In: 2018 Eighth International Conference on Information Science and Technology (ICIST), pp. 342–346. IEEE (2018)
8. Jose, N., Raj, R., Adithya, P., Sivanadan, K.: Classification of forearm movements from sEMG time domain features using machine learning algorithms. In: 2017 IEEE Region 10 Conference, TENCON 2017, pp. 1624–1628. IEEE (2017)
9. Zhengyi, L., Hui, Z., Dandan, Y., Shuiqing, X.: Multimodal deep learning network based hand ADLs tasks classification for prosthetics control. In: 2017 International Conference on Progress in Informatics and Computing (PIC), pp. 91–95. IEEE (2017)
10. Cene, V.H., dos Santos, R.R., Balbinot, A.: Using Antonyan Vardan transform and extreme learning machines for accurate sEMG signal classification. In: 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 5224–5227. IEEE (2018)
11. Gijsberts, A., Atzori, M., Castellini, C., Müller, H., Caputo, B.: Movement error rate for evaluation of machine learning methods for sEMG-based hand movement classification. IEEE Trans. Neural Syst. Rehabil. Eng. **22**(4), 735–744 (2014)
12. Cote-Allard, U., Fall, C.L., Campeau-Lecours, A., Gosselin, C., Laviolette, F., Gosselin, B.: Transfer learning for sEMG hand gestures recognition using convolutional neural networks. In: 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 1663–1668. IEEE (2017)
13. Xing, K., et al.: Hand gesture recognition based on deep learning method. In: 2018 IEEE Third International Conference on Data Science in Cyberspace (DSC), pp. 542–546. IEEE (2018)
14. Zhou, S., Yin, K., Liu, Z., Fei, F., Guo, J.: sEMG-based hand motion recognition by means of multi-class adaboost algorithm. In: 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 1056–1061. IEEE (2017)
15. Geng, W., Du, Y., Jin, W., Wei, W., Hu, Y., Li, J.: Gesture recognition by instantaneous surface EMG images. Sci. Rep. **6**, 36571 (2016)
16. The Ninapro Team: Ninaweb. http://ninapro.hevs.ch/. Accessed 25 Feb 2019