IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 18, 2021, accepted February 6, 2021, date of publication February 15, 2021, date of current version April 7, 2021.
Digital Object Identifier 10.1109/ACCESS.2021.3059499

Surface-Electromyography-Based Gesture Recognition Using a Multistream Fusion Strategy

ZHOUPING CHEN, JIANYU YANG, AND HUALONG XIE
School of Mechanical Engineering and Automation, Northeastern University, Shenyang 110000, China
Corresponding author: Jianyu Yang (jyyang@mail.neu.edu.cn)

This work was supported in part by the National Science Foundation of China under Grant 51505072.

ABSTRACT Gestures are an important way to conduct human-computer interaction. The key problem of gesture recognition depending on sEMG (surface electromyography) is how to achieve high recognition accuracy when there are many types of gestures to classify. To solve this problem, first, two basic models were constructed. One is the ConvEMG model based on dense connectivity, the Inception module and depthwise separable convolution; and the other is the LSTMEMG model based on a bidirectional LSTM (Long Short-Term Memory). Then, the basic models were improved with a multistream fusion strategy which utilizes the correlation between gestures and muscles and the complementary advantages of models. To facilitate comparison with others' models, the models proposed in this paper were tested on the public dataset NinaPro DB5, and the improved model named MultiConvEMG achieves an accuracy of 92.83% for 41 gestures, which is superior to its counterparts in the literature on the same dataset. In addition, experiments containing signal acquisition and gesture recognition were carried out for further testing and evaluation. Experimental results show that all models can achieve an accuracy of more than 95% for 31 gestures, and these models have their own strengths in accuracy, immediacy or training cost. All models built in the paper support using sEMG for end-to-end recognition, which means that artificial features are not needed in the processes and data augmentation or IMU devices are not relied on. In other words, our models outperform and have lower application costs than many known models.

INDEX TERMS Deep learning, gesture recognition, human computer interaction, surface electromyography.

I. INTRODUCTION

As an important method of human-computer interaction, gestures are widely used in the fields of medical rehabilitation, robot control, sign language translation and others. Improving the gesture recognition accuracy helps to rehabilitate post-stroke patients, improves the quality of life of hand amputees and people with language disorders, and produces better effects in other areas where gestures are used as an interactive medium.

At present, the research on gesture recognition using sEMG has achieved satisfactory results when there are fewer types of gestures to classify. Muhammad *et al.* [1] achieved a gesture recognition accuracy of 97.6% by using a CNN to extract the features of the original sEMG signals of 6 gestures from 7 volunteers. Wu *et al.* [2] constructed the LSTM-CNN neural network on the basis of complementary advantages to recognize 5 gestures and achieved a recognition accuracy of 98.14%. Samadani [3] applied a gradually decreasing learning rate to train a bidirectional LSTM, and the recognition accuracy of the method for 17 gestures on the NinaPro DB2 dataset reached 86.7%.

The high-accuracy recognition of a small number of types of gestures cannot meet the requirements for gesture recognition performance in the field of human-computer interaction. The goal is that more gestures can be accurately recognized in the human-computer interaction process to achieve richer functions. However, the more types of gestures that need to be recognized, the lower the recognition accuracy [2], [4], [5]. In most known methods, when there are many types of gestures to be recognized, the recognition rate is not satisfactory. The LSTM-CNN series neural network constructed by Wu *et al.* [2]. achieved a low recognition accuracy of 61.4% for 18 gestures. Shen *et al.* [4] constructed multiple classifiers

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

and used a stacking mechanism, but the recognition accuracy for 40 gestures was only 72.09%. Geng *et al.* [5] built a neural network for gesture recognition on the basis of a single sEMG frame and achieved a recognition accuracy of 77.8% for 52 gestures by performing majority voting on multiple results within a time window. Sun *et al.* [6] achieved a recognition accuracy of 63.86% for 52 gestures with a generative flow model (GFM). A few models relying on auxiliary inertial sensors and feature engineering can achieve high recognition accuracy. For example, the multiview deep learning method proposed by Wei *et al.* [7] requires the careful construction of artificial features. With the aid of inertial sensor data, the recognition rate for 41 gestures in NinaPro DB5 reached 91.31%. Currently, the model with the highest recognition accuracy on DB5 is that of Josephs *et al.* [8] This model also uses inertial sensors to achieve 92% recognition accuracy for 41 gestures.

Some researchers have successfully optimized the recognition method by constructing artificial features [9]–[12]. However, when feature engineering is applied to feature extraction, important information in the signal will inevitably be missed [4]. Additionally, feature engineering takes considerable time and effort, and the use of inertial sensors increases the costs and involves many requirements related to the health of the user's hands. In order to avoid the above situations, not only is an end-to-end model taking only the simply processed sEMG signals as input necessary, but also an excellent model architecture fully extracting the useful information contained in sEMG signals is critical. First, two basic models named ConvEMG and LSTMEMG were constructed on the basis of some excellent deep learning models, such as Inception-V4 [13], DenseNet [14], Xception [15], ResNet [16], and bidirectional LSTM [17]. Then, according to the hypothesis regarding the correlation between gestures and muscles that modeling the signals of each channel separately is more helpful to recognizing gestures and the different advantages of the two basic models, we establish two multistream fusion strategies to improve the basic model and obtain better models, the MultiConvEMG model and the ConvEMG+MultiLSTMEMG model.

On the NinaPro DB5 dataset, MultiConvEMG can recognize 41 gestures with an accuracy of 92.83% using a 200 ms time window. This accuracy exceeds that of other known models tested on the NinaPro DB5 dataset. In order to further analyze the comprehensive performance of the models, experiments containing signal acquisition and gesture recognition were conducted. The improved model named MultiConvEMG+MultiLSTMEMG can recognize 30 gestures with an accuracy of 96.90%, whose recognition result illustrates that only a few highly similar gestures are difficult to distinguish effectively. Although other models are slightly lower than MultiConvEMG+MultiLSTMEMG in terms of accuracy, they have advantages in immediacy or training time.

In the remainder of this paper, the second chapter introduces the architecture of the two basic models, the third chapter introduces the methods that improve the model, the fourth chapter introduces the model validation based on the NinaPro DB5 dataset, and the fifth chapter introduces the experiment based on own data.

II. BASIC MODELS

The CNN and RNN have different advantages related to their modeling abilities. The CNN is better at feature extraction while the RNN is better at time series modeling. Two basic models, one based on a CNN and the other based on an RNN, were constructed to prepare the improved models.

A. CONVEMG ARCHITECTURE

In recent years, some excellent deep learning models have been developed, such as Inception-V4, DenseNet, Xception and ResNet. Inspired by these models, ConvEMG was constructed, which is composed of A module, B module, and a classifier module. The architecture of ConvEMG is shown in FIG. 1.

DenseNet, which uses dense connectivity and encourages feature reuse, outperforms ResNet. The application of dense connectivity and feature reuse not only alleviates the gradient loss, but it also enhances feature propagation. The architecture of A module refers to DenseNet, but the difference between A and B is that A uses different sized convolution kernels to extract features. According to the research of Yue [18], in the feature extraction layer (in the ConvEMG, the feature extraction layer is composed of A module and B module), the higher the proportion of the high-level parameters to the total model parameters, the better the model performance. We set the K values of the three A modules to 16, 32, and 64, respectively, so that the parameters are increased from the lower level to the higher level. In the model, the three A modules are connected via dense connectivity, which means that the input of any module includes the feature map of the outputs of all previous modules and the feature map of the input of the model.

If all feature extraction layers use A module, the feature dimension of the input of the classifier module will be too high, causing two problems. One problem is the risk of overfitting due to the whole model having too many parameters, and the other problem is that model performance decreases as the ratio of the parameters of the layer used for feature extraction to the total parameters of the model decreases. In order to avoid the above situation, B module was built based on Inception-V4. B module uses residual connections instead of dense connectivity to ensure the effective back propagation of the neural network.

At the end of the model is a classifier module, which first uses a maximum pooling layer for downsampling, then reduces the dimension through three one-dimensional convolutional layers, and finally connects a global average pooling and SoftMax activation function.

In the entire model, the depth separable convolution of Xception was used to reduce the number of model parameters when the convolution kernel size was greater than 1.

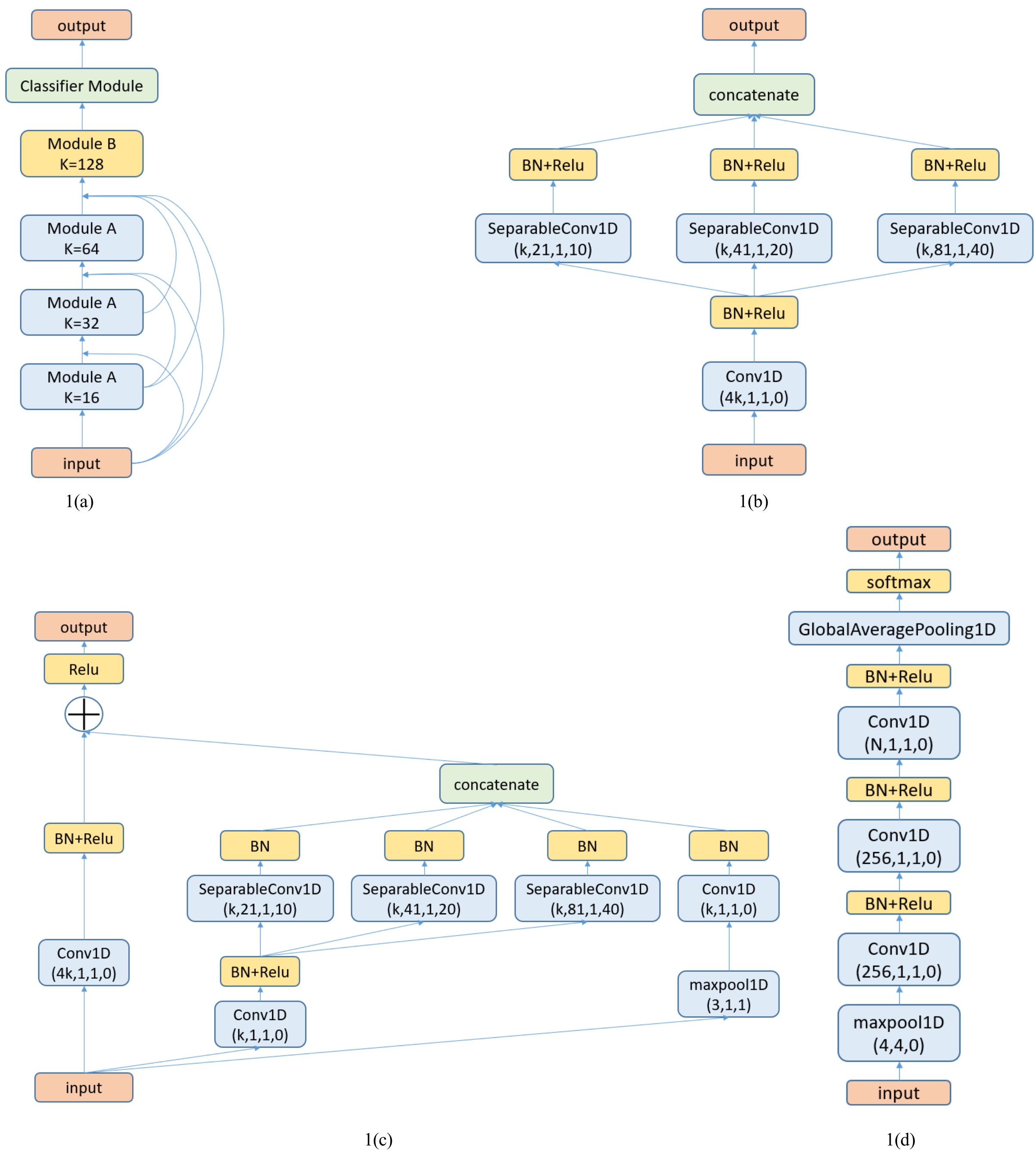


FIGURE 1. **ConvEMG model. (a) is the macroscopic diagram of the model, (b) is the architecture of A module, (c) is the architecture of B module, (d) is the architecture of Classifier module. SeparableConv1D(i,j,k,l) means that the number of filters is i, the size of the kernel is j, the step size is k, and the number of padding is l. Conv1D(i,j,k,l) means that the number of filters is i, the size of the kernel is j, the step size is k, and the number of padding is l; BN+Relu means batch normalization [19] first, then apply the Relu activation function [20]; N is equal to the number of gesture types to be classified.**

Depth separable convolution divides the regular convolution operation into two processes: one is the depthwise convolution, which convolves different channels with different convolution kernels; and the other is the pointwise convolution, which performs regular convolution with a convolution kernel size of 1 on the depthwise convolution result.

B. LSTMEMG ARCHITECTURE

The LSTM module includes a bidirectional LSTM A modulend a classifier module, as shown in FIG. 2.

The bidirectional LSTM module is based on bidirectional LSTM. LSTM [21] can predict only the output with past information; however, in some situations, the output at the

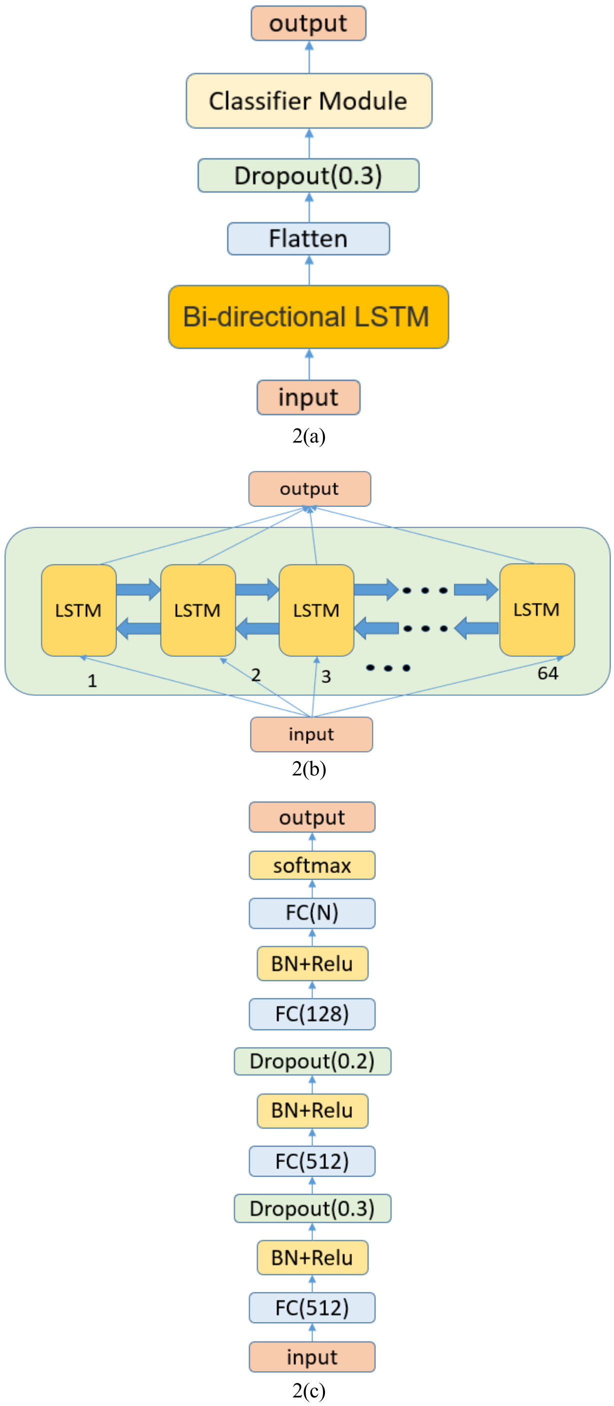


FIGURE 2. LSTMEMG model. (a) is the macroscopic diagram of the model, (b) is a schematic diagram of a bidirectional LSTM, (c) is the architecture of Classifier Module. FC(m) means that the number of neurons is m; N is equal to the number of gesture types to be classified.

current time is not only related to the previous state, but it may also be related to the future state. Bidirectional LSTM [17] can make judgments by combining past and future information. The bidirectional LSTM in the bidirectional LSTM module is used to return the entire output sequence, and the output of all LSTM units is input to the classifier module to obtain the recognition result.

III. MODEL IMPROVEMENT

In this chapter, a multistream strategy, the correlation between gestures and muscles, and their application to improve the provided basic model are introduced.

A. MULTISTREAM FUSION

In recent years, a series of multistream fusion deep learning methods have been proposed in the field of pattern recognition. These methods use multibranch neural networks to model the information from different sensors, different spaces or different times. According to Atrey *et al.* [22], multistream fusion methods can be divided into three categories:

(1) The feature-level fusion of multiple branch features. The application cases are as follows: He *et al.* [23] used a bidirectional LSTM network and an MLP (multilayer perceptron) to extract the features of sEMG signals and then performed feature-level fusion through concatenation, and Eitel *et al.* [24] constructed a dual-stream fusion convolutional neural network to address the problem of object detection from RGB-D images. One branch performs feature extraction on RGB images, and the other branch performs feature extraction on depth images. Finally, the two branches are fused through concatenation.

(2) The decision-level fusion of the classification results of multiple branches. The application case is as follows: Geng *et al.* [5] built a model based on a single sEMG frame for gesture recognition, and the model applies majority voting to multiple results within a time window to obtain the final result.

(3) Hybrid fusion containing both feature-level fusion and decision-level fusion. The application case is as follows: based on the NinaPro dataset, Josephs *et al.* [8] used feature engineering to generate multitype feature maps as the input of the neural network branches and adopted both feature-level fusion and decision-level fusion to construct a model.

B. THE CORRELATION BETWEEN GESTURES AND MUSCLES

According to Jung *et al.* [25], in the human forearm, there are 6 muscle groups that dominate hand movements. Hand movement is accomplished via the coordination of multiple muscle groups. Amma *et al.* [26] used high-density sEMG electrodes to record the sEMG signals at the human forearm and found that only a part of the forearm muscles participate in the movement for a specific gesture and that the muscles participating in the movement vary depending on the type of gesture.

In summary, when there are multiple types of gestures in a time period, gestures are always the result of muscle coordination, but there are no two muscles always working together.

C. STRATEGY FOR MODEL IMPROVENT

In general, sparse EMG electrodes are placed at muscles. Considering the correlation between gestures and muscles,

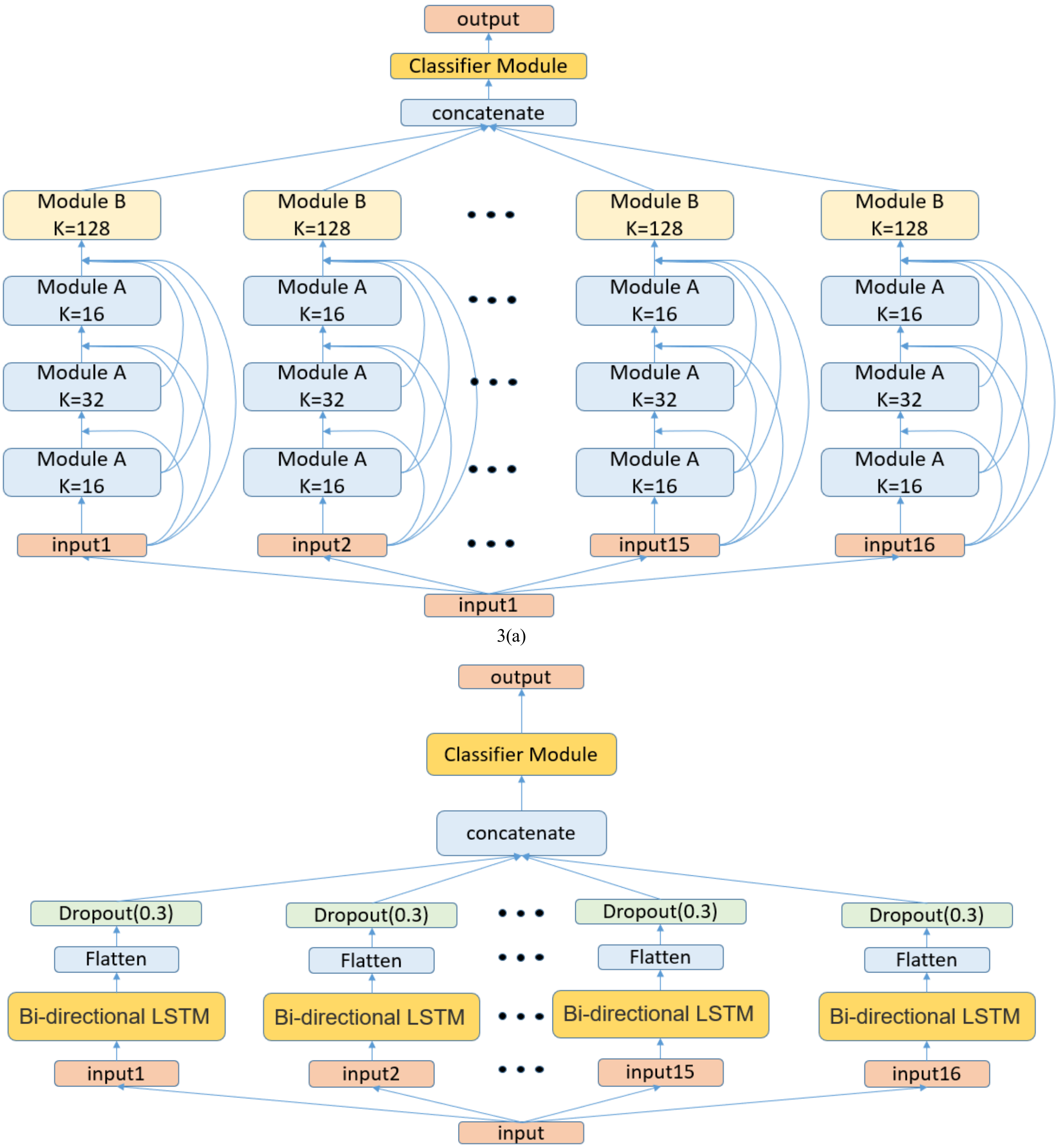


FIGURE 3. **MultiConvEMG model and MultiLSTMEMG model. (a) is MultiConvEMG model, (b) is MultiLSTMEMG model.**

we assumed that taking the signal of each sensor as an independent individual to construct a multibranch neural network is better than the traditional modeling method, which takes the signals collected by all sensors as a whole to construct the model. Based on the above assumptions, an improved strategy was obtained.

Improvement strategy 1: Modules with the same architecture are used to extract the signal features of each sensor separately, and feature-level fusion is performed through feature concatenation to build a new model.

By applying improvement strategy 1 to the two basic models, MultiConvEMG and MultiLSTMEMG were obtained. The architectures of the models are shown in FIG. 3.

In addition, LSTM and CNN have different principles and strengths. Decision-level fusion was used to combine their advantages, and improved strategy 2 was obtained.

Improvement strategy 2: The average of the output results of two different principal models is taken as the final result, as shown in FIG. 4.

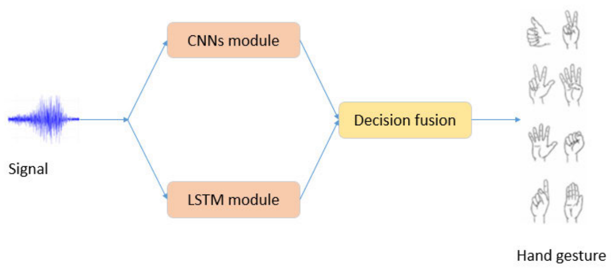


FIGURE 4. Improvement strategy 2.

Using strategy 2, four new models were obtained: ConvEMG+LSTMEMG, MultiConvEMG+LSTMEMG, ConvEMG+MultiLSTMEMG, and MultiConvEMG+MultiLSTMEMG.

IV. MODEL VALIDATION BASED ON NINAPRO DB5

There is no doubt that the recognition accuracy is seriously affected by the signal quality. In order to facilitate comparison with others' models, our model is validated on a public dataset NinaPro DB5. This chapter uses four subsections to provide the experimental details and results. The first section introduces the dataset used for model verification, including the sEMG sensor used in the collection process and the gestures involved in the dataset. In the second section, the data processing methods including the signal processing methods and the data normalization methods are introduced. The details of the model training including the division of the training set and test set, the model training methods and the experimental conditions are presented in the third section. Finally, the fourth chapter presents the experimental results and the comparison with other models validated using the NinaPro DB5 in recent years.

A. DATASET

The Myo armband made by Thalmic Labs is an sEMG sensor with a low cost of only \$200. The MYO armband has 8 channels for signal acquisition using dry electrodes, and its sampling frequency is 200 Hz. Other research-grade sensors used to collect sEMG signals rely heavily on professional knowledge to determine and calibrate the electrode positions, which prevents gesture recognition from being more widely used in production and life.

NinaPro [27] is the largest data collection project in the field of gesture recognition based on sEMG. The project contains 10 large datasets that use multiple sensors to collect data from amputees and complete subjects. The sEMG signals in NinaPro DB5 [28] were collected from two MYO armbands. This dataset contains three subsets: exercise A, exercise B, and exercise C. Exercise A contains 12 basic finger movements and relaxed states; exercise B contains 8 hand extension movements, 9 basic wrist movements and relaxed states; and exercise C contains 23 grasping and functional movements and relaxed states. The dataset was collected from 10 volunteers.

B. DATA PROCESSING

The signal processing method is consistent with that of the dataset publisher Pizzolato *et al.* [28]. That is, first, the original sEMG signal is rectified, and then the rectified signal is filtered with a first-order low-pass Butterworth filter. Finally, a 200 ms time window with a 100 ms overlap is used to divide the signal, as shown in FIG. 5. If there are multiple types of gestures in a window, the label of the window is determined by the result of the majority voting.

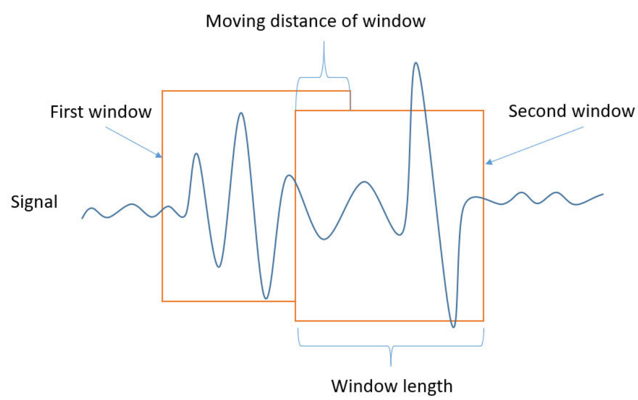


FIGURE 5. Single channel signal windowing.

The processed signal needs to be u-law converted before it's input to the deep learning model [29]. This normalization method was recently applied to the field of gesture recognition based on sEMG signals by Rahimian *et al.* [30], showing equally effective and superior to traditional min-max normalization, however it was generally used in the field of speech and communication. The specific formula is shown in (1). U is 256.

$$F\left(x_t\right) = sign(x_t)(\ln(1+u|x|)/\ln\left(1+u\right)) \tag{1}$$

The change of the signal waveform during the whole process is shown in the FIG. 6.

C. MODEL TRAINING

The division of the training set and test set is also the same as in Eitel *et al.* [24], that is, the data of the third repetition and fifth repetition in exercise B and exercise C are used in testing, and the others are used in training.

ConvEMG and MultiConvEMG were trained for 200 epochs with the Adam optimizer with a learning rate of 0.0001. LSTMEMG and MultiLSTMEMG adopt the SGD optimizer. The learning rate of the SGD optimizer combines the step-down rate with cyclical learning rates [31]. In one cycle, the learning rate is 0.1 for epochs 1 to 3, 0.01 for epochs 4 to 13, 0.001 for epochs 14 to 23, and 0.0001 for epochs 24 to 35. Five cycles are performed.

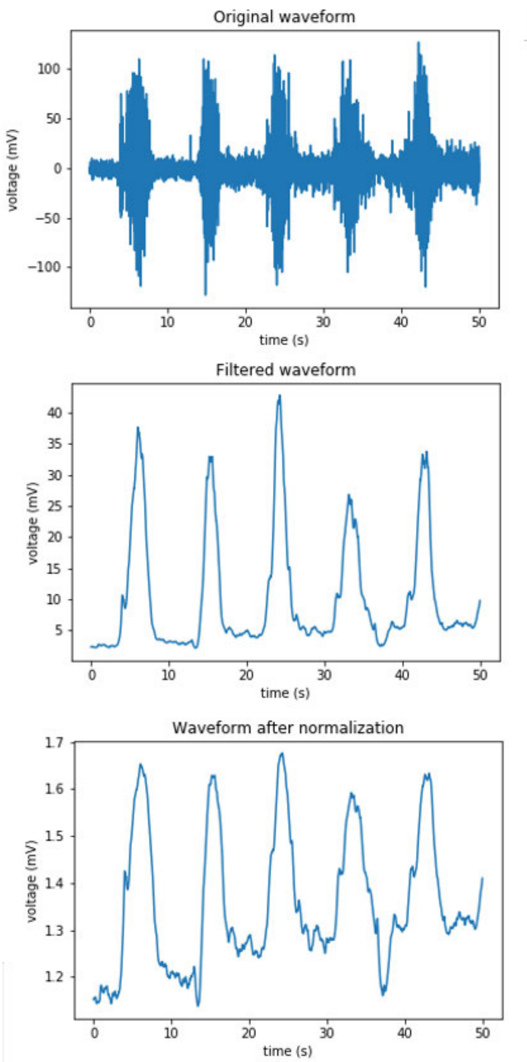


FIGURE 6. Changes in waveform.

To perform our training and tests, an AMD3800X CPU and an RTX2070s GPU with the Keras deep learning framework were used.

D. RESULTS

The average recognition accuracy of ten volunteers was taken as the accuracy of the model. The accuracies of all models in this paper are shown in TABLE 1, and the model with the highest accuracy is marked with black. It can be seen that the two models applying improvement strategy 1 are better than the basic model, and MultiConvEMG has the highest recognition rate among all models. Among the models applying improvement strategy 2, only ConvEMG+MultiLSTMEMG exceeds the sub models using decision-level fusion.

The summary of the gesture recognition researches based on NinaPro DB5 that has been conducted in recent years is shown in TABLE 2. The existing model proposed by other researchers with the highest recognition rate is the model proposed by Josephs *et al.* [8]. The recognition accuracy of the MultiConvEMG model is slightly higher than that of the model proposed by D. Josephs *et al.* MultiConvEMG and ConvEMG+MultiLSTMEMG are far superior to other models that use only sEMG for gesture recognition.

TABLE 1. The accuracies of all models proposed in the paper.

Model	Accuracy
ConvEMG	90.80%
LSTMEMG	88.11%
MultiConvEMG	**92.83%**
MultiLSTMEMG	90.70%
ConvEMG+LSTMEM	90.40%
MultiConvEMG+LSTMEMG	91.44%
ConvEMG+MultiLSTMEMG	91.70%
MultiConvEMG+MultiLSTMEMG	92.56%

TABLE 2. Models verifying on ninapro DB5.

Model	Types of gestures	Window length	Accuracy
MultiConvEMG	Exercise B, Exercise C	200ms	**92.83%**
S. Pizzolato et al. [28] (2017)	Exercise B, Exercise C	200ms	69.13%
S. Shen et al. [4] (2019)	Exercise B, Exercise C	200ms	72.09%
W. Wei et al. [7] (2019) with IMU	Exercise B, Exercise C	200ms	91.31%
W. Wei et al. [7] (2019)	Exercise B, Exercise C	200ms	90.00%
D. Josephs et al. [8] (2020) with IMU	Exercise B, Exercise C	260ms	92%
U. Cote-Allard et al. [32] (2019)	Exercise B	260ms	68.98%
Y. Wu et al. [2] (2019)	Exercise B	300ms	61.4%
L. Chen et al. [33] (2020)	Exercise B	260ms	67.42%

V. EXPERIMENT BASED ON OWN DATA

In order to further evaluate and test the comprehensive performance of the models, experiments containing signal acquisition and gesture recognition were conducted. To introduce the experiment, three subsections are used in this section. The first subsection introduces the process of signal acquisition. In the second subsection, modeling process are introduced. The results of experiment and results-based analysis are presented in the third subsection.

A. SIGNAL ACQUISITION

This experiment uses self-developed device to collect sEMG. The device collecting sEMG is shown in FIG. 7. The device, contains eight differential acquisition channels with 360 Hz sampling frequency. The electrodes were sticked at the the shown positions of the upper and lower arm armbands. The collected objects are 30 gestures of the left hand of a healthy 26-year-old man, as shown in FIG.8. The entire collection process is the same as the previous section.

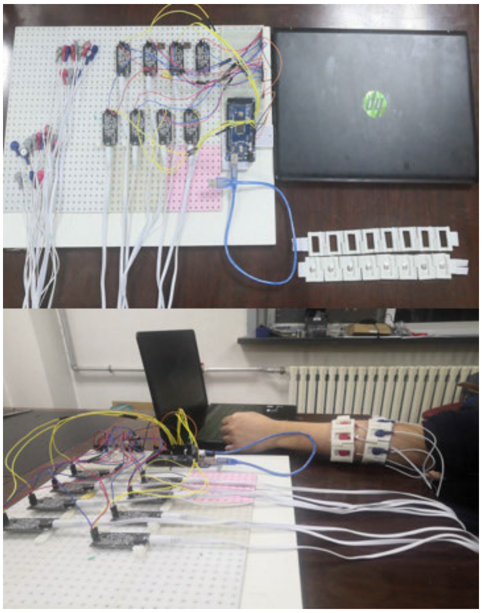


FIGURE 7. Experimental device.

B. MODELING PROCESS

The modeling process of experiment is the same as the previous chapter except for signal filtering. In this experiment, mean filtering is used to calculate the mean value of all points within 100ms as the actual value of the point (In order to perform mean filtering, 36 zeros are added before each signal). This method is better than Butterworth filtering in terms of real-time performance, which is more conducive to the use of the model in real environments.

C. RESULTS

In view of the fact that the actual application of the model must consider not only the accuracy rate but also immediacy and training time. Also, the models and the model improvement strategies should be comprehensively evaluated and tested.

The accuracy of the models are shown in TABLE 3. All models in the paper can achieve higher than 95% accuracy. MultiConvEMG+MultiLSTMEMG is the model with the highest accuracy among all the models, reaching 96.90%. More over all models obtained through improved strategies perform better than the basic model. Although other models are slightly lower than MultiConvEMG+MultiLSTMEMG in terms of accuracy, they have better results in immediacy (inference time is mainly considered) or training time.

FIG. 9 is drawn to evaluate the model and improvement strategy more easily. The x-axis is the increase in accuracy of the improved model compared to the model with the highest accuracy in the basic model. The y-axis is the increase in time of the improved model relative to the model with the lowest training time or lowest inference time in the basic model. The larger the y value and the smaller the x value indicate that the corresponding improvement strategy would be more cost-effective, which provides reference for choosing suitable improvement strategy.

Compared with the average recognition rate, the recognition accuracy of each gesture in the model is also worthy

FIGURE 8. Gestures and its labels.

TABLE 3. Experimental results.

Model	Accuracy	Inference time	Training time
C (ConvEMG)	95.53%	**0.13s**	15s
L (LSTMEMG)	95.05%	0.19ms	**1s**
MC (MultiConvEMG)	96.47%	0.90ms	102s
ML (MultiLSTMEMG)	96.03%	1.1ms	7s
C+L (ConvEMG+LSTMEM)	95.94%	0.32ms	16s
MC+L (MultiConvEMG+LSTMEMG)	96.62%	1.1ms	103s
C+ML (ConvEMG+MultiLSTMEMG)	96.10%	1.2ms	22s
MC+ML (MultiConvEMG+ MultiLSTMEMG)	**96.90%**	2.0m s	109s

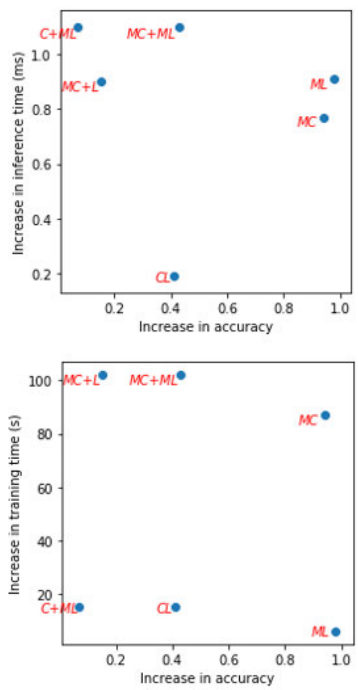


FIGURE 9. Gestures and its labels.

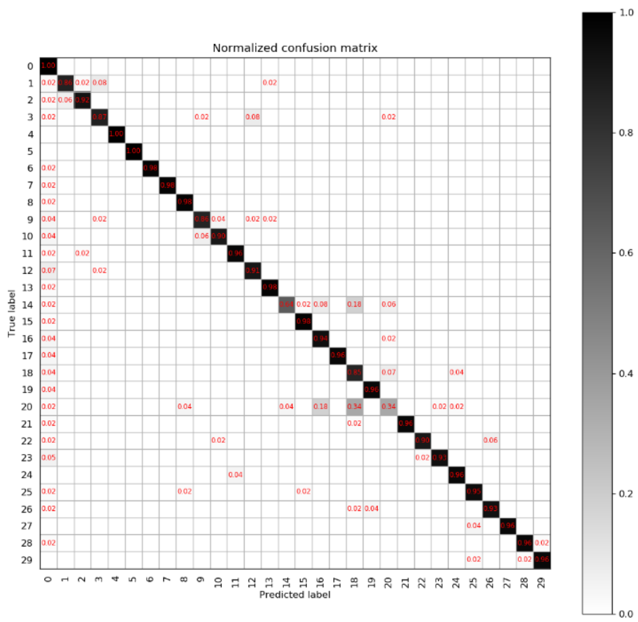


FIGURE 10. The recognition accuracy of each gesture.

of attention. MultiConvEMG+MultilSTMEMG is the model with the highest recognition accuracy. The recognition accuracy of each gesture is shown in Figure 10, which indicates that only a few highly similar gestures are difficult to recognize effectively, and the overall recognition effect is satisfactory.

VI. CONCLUSION

The work and contributions of this paper are as follows:

1. High-performance end-to-end models were built. The models in this paper do not use feature engineering or inertial sensors. Furthermore, a 92.83% recognition accuracy was achieved for 41 gestures from the NinaPro DB5 dataset, exceeding the recognition accuracy of other models that use feature engineering and inertial sensor; and this is the highest accuracy among the studies using this dataset.

2. Excellent model architectures were constructed. The architecture of the models in this paper incorporated the advantages of existing models and followed some rules of thumb.

3. A modeling method that can be extended to other fields was proposed. In this paper, first, sEMG was used as a general timing signal to build two basic models, and then the human hand movement mechanism and the performance characteristics of the models were combined to improve the models. For pattern recognition problems, the modeling method that generalized basic models are constructed first and then the basic models are improved according to the laws existing in the field of the problem and the characteristics of the basic model can be extended to other fields.

REFERENCES

[1] M. Z. U. Rehman, A. Waris, S. Gilani, M. Jochumsen, I. Niazi, M. Jamil, D. Farina, and E. Kamavuako, ''Multiday EMG-based classification of hand motions with deep learning techniques,'' *Sensors*, vol. 18, no. 8, p. 2497, Aug. 2018.

[2] Y. Wu, B. Zheng, and Y. Zhao, ''Dynamic gesture recognition based on LSTM-CNN,'' in *Proc. Chin. Automat. Congr. (CAC)*, 2019, pp. 2446–2450.

[3] A. Samadani, ''Gated recurrent neural networks for EMG-based hand gesture classification. A comparative study,'' in *Proc. 40th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2018, pp. 1–4.

[4] S. Shen, K. Gu, X.-R. Chen, M. Yang, and R.-C. Wang, ''Movements classification of multi-channel sEMG based on CNN and stacking ensemble learning,'' *IEEE Access*, vol. 7, pp. 137489–137500, 2019.

[5] W. Geng, Y. Du, W. Jin, W. Wei, Y. Hu, and J. Li, ''Gesture recognition by instantaneous surface EMG images,'' *Sci. Rep.*, vol. 6, no. 1, Dec. 2016, Art. no. 36571.

[6] W. Sun, H. Liu, R. Tang, Y. Lang, J. He, and Q. Huang, ''SEMG-based hand-gesture classification using a generative flow model,'' *Sensors*, vol. 19, no. 8, p. 1952, Apr. 2019.

[7] W. Wei, Q. Dai, Y. Wong, Y. Hu, M. Kankanhalli, and W. Geng, ''Surface-electromyography-based gesture recognition by multi-view deep learning,'' *IEEE Trans. Biomed. Eng.*, vol. 66, no. 10, pp. 2964–2973, Oct. 2019.

[8] D. Josephs, C. Drake, A. Heroy, and J. Santerre, ''sEMG gesture recognition with a simple model of attention,'' in *Proc. Mach. Learn. Health NIPS Workshop*, 2020, pp. 126–138.

[9] Y. Sun, C. Xu, G. Li, W. Xu, J. Kong, D. Jiang, B. Tao, and D. Chen, ''Intelligent human computer interaction based on non redundant EMG signal,'' *Alexandria Eng. J.*, vol. 59, no. 3, pp. 1149–1157, Jun. 2020.

[10] G. Li, J. Li, Z. Ju, Y. Sun, and J. Kong, ''A novel feature extraction method for machine learning based on surface electromyography from healthy brain,'' *Neural Comput. Appl.*, vol. 31, no. 12, pp. 9013–9022, Dec. 2019.

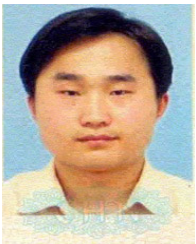
[11] S. Ying *et al.*, ''Gesture recognition algorithm based on multi-scale feature fusion in RGB-D images,'' *IET Image Process.*, vol. 14, no. 15, pp. 3662–3668, Dec. 2020.

[12] A. C. Turlapaty and B. Gokaraju, ''Feature analysis for classification of physical actions using surface EMG data,'' *IEEE Sensors J.*, vol. 19, no. 24, pp. 12196–12204, Dec. 2019.

[13] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," in *Proc. AAAI Conf. Artif. Intell.*, 2016, pp. 1–7.
[14] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," 2016, *arXiv:1608.06993*. [Online]. Available: https://arxiv.org/abs/1608.06993
[15] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," 2016, *arXiv:1610.02357*. [Online]. Available: http://arxiv.org/abs/1610.02357
[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
[17] A. Graves, "Supervised sequence labelling with recurrent neural networks," *Stud. Comput. Intell.*, vol. 385, 2012.
[18] X. Yue, "The influence of the amount of parameters in different layers on the performance of deep learning models," *Comput. Appl. Softw.*, vol. 5, no. 12, pp. 445–453, 2015.
[19] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
[20] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1097–1105.
[21] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[22] P. K. Atrey, M. A. Hossain, A. El Saddik, and M. S. Kankanhalli, "Multimodal fusion for multimedia analysis: A survey," *Multimedia Syst.*, vol. 16, no. 6, pp. 345–379, Nov. 2010.
[23] 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.
[24] A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller, and W. Burgard, "Multimodal deep learning for robust RGB-D object recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Jul. 2015, pp. 5636–5639.
[25] P.-G. Jung, G. Lim, S. Kim, and K. Kong, "A wearable gesture recognition device for detecting muscular activities based on air-pressure sensors," *IEEE Trans. Ind. Informat.*, vol. 11, no. 2, pp. 485–494, Apr. 2015.
[26] C. Amma, T. Krings, J. B. Er, and T. Schultz, "Advancing muscle-computer interfaces with high-density electromyography," in *Proc. 33rd Annu. ACM Conf. Hum. Factors Comput. Syst.*, 2015, pp. 929–938.
[27] M. Atzori and H. Muller, "The ninapro database: A resource for sEMG naturally controlled robotic hand prosthetics," in *Proc. 37th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Aug. 2015, pp. 7151–7154.
[28] 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.
[29] *Pulse Code Modulation (PCM) of Voice Frequencies*, document ITU-T Rec. G.711, 1988.
[30] E. Rahimian, S. Zabihi, S. F. Atashzar, A. Asif, and A. Mohammadi, "XceptionTime: A novel deep architecture based on depthwise separable convolutions for hand gesture classification," 2019, *arXiv:1911.03803*. [Online]. Available: https://arxiv.org/abs/1911.03803
[31] L. Smith, "Cyclical learning rates for training neural networks," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, 2017.
[32] U. Cote-Allard, C. L. Fall, A. Drouin, A. Campeau-Lecours, C. Gosselin, K. Glette, F. Laviolette, and B. Gosselin, "Deep learning for electromyographic hand gesture signal classification using transfer learning," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 4, pp. 760–771, Apr. 2019.
[33] L. Chen, J. Fu, Y. Wu, H. Li, and B. Zheng, "Hand gesture recognition using compact CNN via surface electromyography signals," *Sensors*, vol. 20, no. 3, p. 672, Jan. 2020.

ZHOUPING CHEN is currently pursuing the master's degree in mechanical engineering under the supervision of Dr. Jianyu Yang. His research interest includes human–computer interaction.

JIANYU YANG is currently an Assistant Professor of mechanical engineering. He is also a Doctor of mechanical engineering. His research interests include the hand exoskeleton, robotics and industrial automation, and numerical manufacturing.



HUALONG XIE is currently an Associate Professor of mechanical engineering. He is also a Doctor of information science. He is working on developing prosthetic and exoskeleton devices for human low limbs, sEMG based human motion recognizing, and so on.