

# Similarity Function for One-Shot Learning to Enhance the Flexibility of Myoelectric Interfaces

Xiang Wang<sup>1</sup>, Xu Zhang<sup>1</sup>, Member, IEEE, Xiang Chen<sup>1</sup>, Member, IEEE, Xun Chen<sup>1</sup>, Senior Member, IEEE, Zhao Lv<sup>1</sup>, and Zhen Liang

**Abstract—Objective:** This study aims to develop a flexible myoelectric pattern recognition (MPR) method based on one-shot learning, which enables convenient switching across different usage scenarios, thereby reducing the re-training burden. **Methods:** First, a one-shot learning model based on a Siamese neural network was constructed to assess the similarity for any given sample pair. In a new scenario involving a new set of gestural categories and/or a new user, just one sample of each category was required to constitute a support set. This enabled the quick deployment of the classifier suitable for the new scenario, which decided for any unknown query sample by selecting the category whose sample in the support set was quantified to be the most like the query sample. The effectiveness of the proposed method was evaluated by experiments conducting MPR across diverse scenarios. **Results:** The proposed method achieved high recognition accuracy of over 89% under the cross-scenario conditions, and it significantly outperformed other common one-shot learning methods and conventional MPR methods ( $p < 0.01$ ). **Conclusion:** This study demonstrates the feasibility of applying one-shot learning to rapidly deploy myoelectric pattern classifiers in response to scenario change. It provides a valuable way of improving the flexibility of myoelectric interfaces toward intelligent gestural control with extensive applications in medical, industrial, and consumer electronics.

**Index Terms—**Myoelectric control, electromyogram (EMG), one-shot learning, cross-scenario, flexibility.

## I. INTRODUCTION

AN ELECTROMYOGRAM (EMG) represents an electrophysiological signal reflecting muscular activities that

Manuscript received 14 June 2022; revised 29 November 2022; accepted 5 February 2023. Date of publication 6 March 2023; date of current version 22 March 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62271464 and in part by the Natural Science Research Project of Colleges and Universities in Anhui Province under Grant KJ2020ZD16. (Corresponding authors: Xu Zhang; Zhen Liang.)

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Clinical Medicine Research Ethics Committee of the First Affiliated Hospital of Anhui Medical University (AHMU; Hefei, Anhui China) and the Ethics Review Board of the University of Science and Technology of China (USTC; Hefei, Anhui, China).

Xiang Wang and Zhen Liang are with Department of Biomedical Engineering, Anhui Medical University, Hefei 230022, China (e-mail: zliang@ahmu.edu.cn).

Xu Zhang, Xiang Chen, and Xun Chen are with the School of Microelectronics, University of Science and Technology of China, Hefei 230027, China (e-mail: xuzhang90@ustc.edu.cn).

Zhao Lv is with the Institute of Physical Science and Information Technology, Anhui University, Hefei 230601, China.

Digital Object Identifier 10.1109/TNSRE.2023.3253683

formulate movements under the control of motor nerves. It can be used to decode and understand movement intentions and to provide useful commands for controlling externally powered devices, such as prosthetic and orthotic robots [1], [2]. EMG has been used in the well-known technique termed myoelectric control, where the surface EMG (sEMG) is mainly employed by placing electrodes over the skin surface to sense movements in a non-invasive manner [3], [4]. Myoelectric pattern recognition (MPR) is a ground-breaking technology that enables dexterous control of multi degrees of freedom easily [5], [6]. There are also many commercialized MPR products [7], [8], [9], [10] that are designed for prosthetic control and rehabilitation treatment [6], [11]. Besides, the MPR technology has significantly broader applications and can exhibit potential in consumer electronics [12], [13]. In recent years, MPR technology has attracted much attention as a novel interface for information input and editing with applications in consumer electronics for motion-sensing games and augmented or virtual reality for education and entertainment [9], [14]. Current interaction scenarios mostly use gestures as control commands [9], [15]. Although satisfactory performance has been reported to demonstrate the effectiveness of the MPR technology in laboratory conditions, its applications may not always be successful in practice [4], [12].

Consumer electronics applications always require high flexibility of the interactive system that maintains the gesture recognition performance across diverse application scenarios involving using different command sets and switching to different users. In a routine implementation of the MPR technology, a classifier is trained on a fixed command set for a specific user. Similarly, a myoelectric classifier is usually built with a specific user's data due to the great cross-user variabilities of the sEMG signals, leading to its application in a user-dependent manner [12]. Therefore, a conventional myoelectric classifier is applicable for a certain scenario depending on a predefined gestural command set and a specific user. When this myoelectric control system is used in another scenario, it suffers from degraded performance or is utterly incompetent because it lacks effective learning and adaptation to new scenarios. The original classifier needs to be re-trained; otherwise it leads to compromised performance. The burden of re-training is a discouraging and frustrating process for the user [16], [17], [18]. Under this situation, the classifier must be re-trained or calibrated with much well-labeled data from the new scenario, imposing a great burden on the user. Therefore, there is a great demand for flexible myoelectric

control interfaces that can be used across diverse application scenarios.

Many studies have been conducted to make the myoelectric control system adaptive to different scenarios. The essence of the cross-scenario myoelectric pattern recognition problem is the different distribution of samples between different scenarios, similar to the domain shift problem in computer vision, and transfer learning is a typical solution. Transfer learning aims to improve the performance of a model on a target domain by transferring knowledge contained in a different but related source domain [19], [20], [21], [22]. Two types of transfer learning have been applied to MPR. One type is supervised transfer learning based on deep neural networks. Ameri et al. [23] and Hu et al. [24] introduced this type to allow the myoelectric classifier to be calibrated towards adaption to electrode shift. Further, Chen et al. [25] proposed different transfer learning models with fine-tuned parameters on a few repetitions to improve the recognition accuracy of new users and new categories. However, these studies failed to eliminate the requirement of large, labeled data in the target domain, still imposing potentially overwhelming re-training burdens on users. The other type is unsupervised transfer learning, where both domain adaption (DA) [26] and domain generalization (DG) [22] have become popular approaches. The difference between DA and DG is that DA has access to the unlabeled target domain data, while DG cannot see them during training. Cote-Allard et al. [10], [14] proposed unsupervised adaptive models, which use DA approaches to maximize the performance of a given new user using existing training source domain(s) to overcome the distribution difference problem across users [14]. Wu et al. [27] used data augmentation, a DG approach, to improve model generalization capability through data augmentation. Although these unsupervised approaches do not impose an additional re-training burden at all, their performance is not satisfactory under conditions with large cross-domain differences due to limited knowledge of the new domain.

To ensure the flexibility of myoelectric interfaces, it is not compulsory to eliminate the requirement for re-training or calibration in new scenarios. If the data required from the user are minimal, for example, just one execution per each gestural pattern, such a fast and simple re-training procedure is acceptable for the user under a new scenario. Especially involving new categories, as far as we know, it is difficult to implement category substitution in the command set completely without any re-training burden [28], [29], [30], [31]. As an alternative transfer learning strategy, one-shot learning is applicable to meet such a requirement. Many previous studies have reported its successful applications in image processing in particular. A facial image recognition system proposed by Chanda et al. [32] based on a one-shot learning approach allowed a new user to be added using only one photo without a complex re-training procedure. Moreover, Koch et al. [33] conducted a one-shot learning approach using Siamese neural networks (SNNs) for image classification to enable the correct identification of images in categories that were not involved in the training dataset at the least cost of re-training. In these studies, using one-shot learning enabled

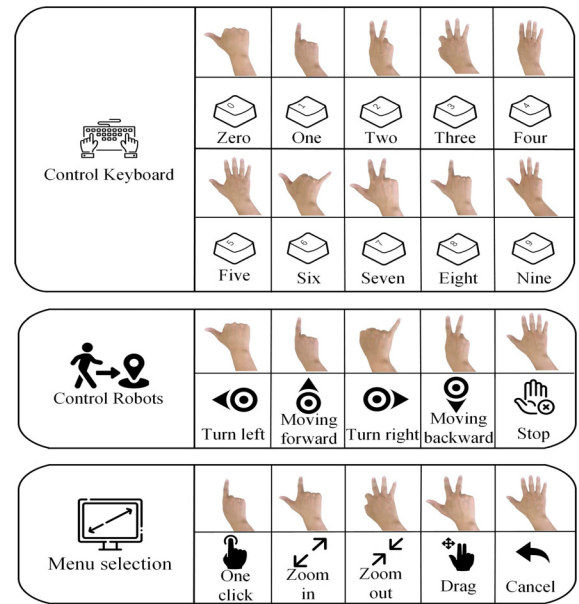


Fig. 1. Illustration of gestural interfaces applied to three different scenarios with different sets of gestural commands.

the fast calibration of the classifier to be suitable for new users and categories, drawing an analogy for developing flexible myoelectric interfaces across multiple scenarios.

Inspired by the advance of one-shot learning, a new method for MPR is proposed in this study to enhance the flexibility of myoelectric interfaces applicable to various application scenarios, including switching to a new user and/or using a different command set. This method relies on the SNN, which can generate a similarity evaluation metric by learning how different any gestural category is from another rather than what each category is. Pioneering studies [31] on using this similarity metric were introduced in the computer vision field for image classification. This network is employed due to its contribution that the features learned in the source domain can be generalized to the target domain, even if the target domain's command set is changed in gestural categories or their number. Implementing the proposed method can substantially reduce the re-training burden when cross-domain manipulations are required. Our work helps improve the myoelectric control systems' application flexibility and enhance their robustness against variabilities in both the command category set and the user.

## II. METHODS

### A. Subjects

Seven healthy and non-disabled subjects (aged 24–35 years; five males and two females, all right-handed) were recruited for this study. The study was approved by both the Clinical Medicine Research Ethics Committee of the First Affiliated Hospital of Anhui Medical University (AHMU, Hefei, Anhui 230022, China) and the Ethics Review Board of the University of Science and Technology of China (USTC, Hefei, Anhui 230026, China). Informed and written consent was obtained from all subjects before they participated in any experiment procedure.

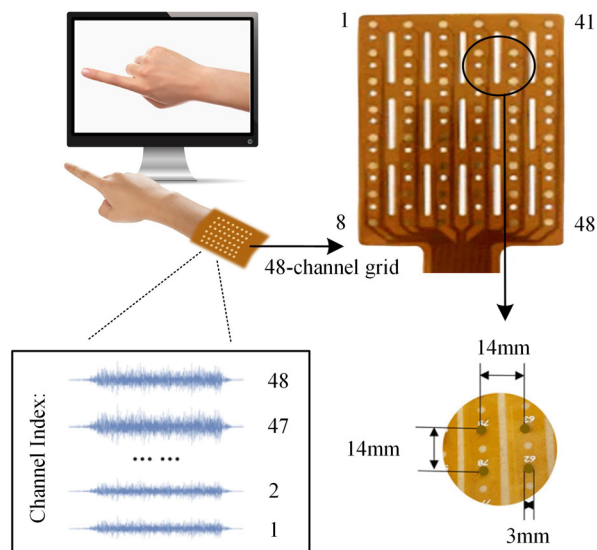


Fig. 2. Placement of the 2-dimensional electrode array in a form of  $6 \times 8$  on the forearm muscles for recording HD-sEMG data.

## B. Experiments

The subjects sat in a height-adjustable chair with their arms on a table during the experiment. Meanwhile, pads with 70% isopropyl alcohol were applied to clean the skin surface of the subjects' tested forearms. A high-density electrode array was placed to mainly cover the posterior side of the tested forearm for recording HD-sEMG data from major forearm extensors. It was designed with 48 electrode probes arranged in a  $6 \times 8$  grid (Fig. 2). The diameter of each electrode probe was 3 mm, and the distance between two neighboring electrodes was 14 mm. In addition, two common reference electrodes were attached to the olecranon of both arms, respectively. Thus, each electrode in the array formulated one recording channel concerning the reference. The HD-sEMG signals were recorded by a custom-made data recording system. The raw signals were amplified with a total gain of 60 dB, band-pass filtered at 20–450 Hz, and digitalized via a 16-bit analog-digital converter at a sampling rate of 1 kHz. The recorded data were transferred to a computer via a USB cable, and software with a graphical user interface (GUI) was developed to monitor and process all the recorded EMG data on the computer screen in real time.

1) *Data Collection*: All subjects were asked to perform ten gestural categories (labeled as G1-G10) involving the extension of different fingers or finger combinations (as the main functions of the finger extensors) (Fig. 3). The subjects were instructed to perform four repetitions of each gestural category at a stable and comfortably medium force level, generally corresponding to 30–40% maximal voluntary contraction of the muscles. After completing each repetition, the hands and fingers slowly returned to a neutral position and remained relaxed, with all fingers naturally bent. A video clip was prepared and played as a guideline to instruct the subjects on implementing these gestures and their timings. Following the video guidance, the subjects were asked to hold each repetition of the gesture for 5 s with a relaxation period of 4 s between two consecutive repetitions. Some of these gestures could be selected to form different gestural command sets suitable for various application scenarios (Fig. 1).

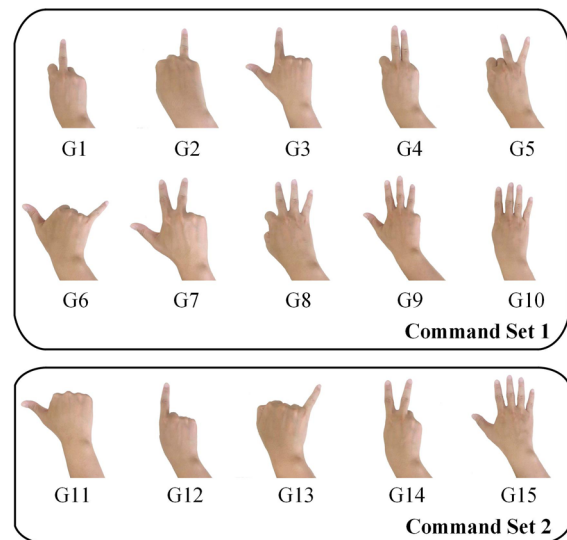


Fig. 3. Illustration of fifteen gestural categories. There are two gestural category sets, namely Command Set 1 including 10 categories (G1 - G10) and Command Set 2 containing 5 categories (G11 - G15), respectively.

In this study, we intentionally selected ten categories (e.g., G3, G6-G12, G14, G15 in Fig. 3) to form a set for simple numeric input. In addition, another combination of five categories (e.g., G11 - G15 in Fig. 3) was formed as the command set for manipulating an industrial robot in different degrees of freedom. Both sets of gestural commands (Fig. 3) were defined in this study to test the rapid customization and deployment of the classifier under the condition of changing the gestural command set by adding or alternating new categories.

2) *Online Testing With New Command Set*: In addition to the above collection, our system supports real-time data processing and testing. The collected data were segmented into a series of data windows with a length of 128 ms and an overlap of 50% (i.e., window increment of 64 ms). These windows were the basic samples for the subsequent MPR, and the specific data processing algorithm was detailed in the following subsection. For each user, the previously acquired data were used as source domain data to pre-train a similarity model (rather than a final classification model). On this basis, a cross-scenario online test was designed: the user was assumed to select a new Command Set 2, as shown in Fig. 3, to be used in a new scenario, including five new categories (i.e., G11-G15). During the calibration stage, subjects followed the on-screen cues and performed each gesture only once, for about a 1-s execution period. A randomly selected sample from the middle segment was used to form the support set. Then, the system combines the previously trained model and completes the construction of the new classifier almost instantaneously. In the testing phase, the system randomly generated a series of gestural tasks for manipulation, displayed in a GUI (Fig. 4), and asked the subjects to follow the on-screen cues to execute each task in turn for 1 s. When a gestural task appeared, each task was displayed for 3 s, followed by another 3-s rest between two executions. During the whole testing phase, each gesture category was tested at least 20 times, regardless of the order of appearance, and the data of up to 20 s for each category was collected (not counting the baseline during the no action period) approximately equal to the total amount of data

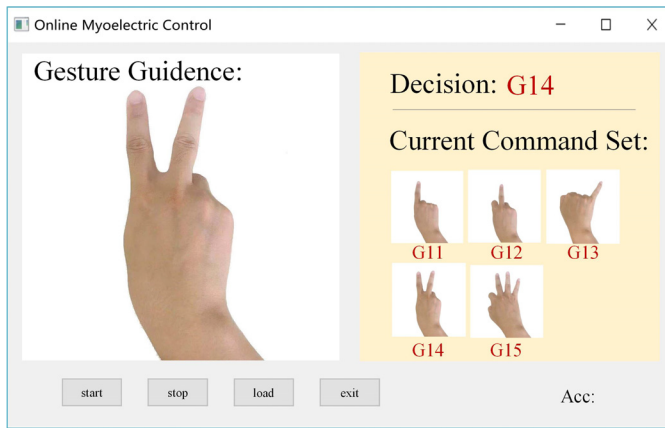


Fig. 4. The user interface of the software used for the online testing experiment. There is a demonstration photo on the left panel to show the gestural task required to be performed by the user. The system decision is reported on the right panel, along with the currently used command set.

for each gesture in the training data (approximately 300 windows could be collected). The testing data was processed in real-time in the system to make instant decisions about gesture recognition and to calculate the recognition accuracy concerning the cues (i.e., the ground truth) displayed to the user. The recognition accuracy was defined as the number of correctly recognized windows divided by the total number of windows to be recognized, where the windows corresponding to the baseline were not considered.

All the data of the testing process were also saved and pooled into the same database with the training data so that the data stored from each subject included 15 gestures with up to 20 s of data per gesture for subsequent retrospective offline data analyses, including testing other algorithms, changing the split of training, testing data, and performing cross-validation.

### C. Data Processing for MPR Using One-Shot Learning

Fig. 5 shows the flowchart of the proposed method for fast calibrating an MPR control system suitable for a new scenario by taking advantage of the similarity prediction capability learned from many sample pairs using one-shot learning. Under a given application scenario with any user and any gestural command set, just one sample of each gestural category is required from the user to form a support set. The classifier suitable for this new scenario can be established consequently with the cue of the support set to maintain the high performance of the MPR control.

1) *Data Segmentation and Feature Extraction*: As described in the experiment, the data were segmented into windows according to time. Then, the part of the windows that belonged to the muscle activation was selected from them. The RMS amplitude thresholding method proposed by Pasinetti et al. [34] was employed for determining muscle activation windows. It was based on detecting both onset and offset timings of every repetition of muscle activation, where the baseline data were discarded accordingly. Thus, each gestural category, including multiple repetitions over about 20s in total, could produce about 300 windows. Then, for the windows from muscle activities during gestural performances (G1-G15), four time-domain (TD) features, originally

proposed by Hudgins et al. [35], (namely mean absolute value (MAV), number of zero-crossings (ZC), number of slope sign change (SSC), and waveform length (WL)) were extracted from each channel of the HD-sEMG data. Therefore, a feature matrix of  $6 \times 8 \times 4$  was formed for each analysis window consisting of 48 channels, where the 2-D electrode array intentionally retained the size of  $6 \times 8$  to maintain its spatial information. We could also regard each feature matrix as a featured image where a sEMG channel represented each pixel in a resolution of  $6 \times 8$  in the array with the same size. Further, our feature image has four color channels represented by four TD features. Each featured image derived from an individual window was considered a basic sample in the pattern recognition analysis in this study.

In addition, a data augmentation approach was applied to all samples of the training dataset described in the following pattern recognition approach as reported in previous studies [27], [36]. A shift operation is performed on each feature image using moving the original image by one pixel in both vertical and horizontal directions while the image size is maintained. Such a transformation resulted in cropping out 13 pixels in the original images while specifying new regions to be filled with the new pixels. These new regions were filled by duplicating the pixels from the edge near the original image. Consequently, the amount of training data samples was doubled compared to its original sample size. However, conducting this data augmentation approach on any sample was unnecessary for calibration or testing.

#### 2) Siamese Neural Network for Pattern Similarity Metric:

Before conducting the pattern classification, we had to build the SNN first. Unlike general neural networks capable of predicting specific patterns of individual samples, the SNN is often used to evaluate whether two samples are similar due to their unique structure [37]. As shown in Fig. 6, there are a pair of sub-networks at the front end, each consisting of three blocks in the same structure and essentially sharing the same weights. Each sub-network can be considered a copy of the other. Two sEMG image samples are required to be fed into both subnetworks simultaneously to obtain their respective feature maps flattened as one-dimensional feature vectors by Block 3. Both resultant feature vectors are then passed through a metric function module consisting of a Euclidean distance function and a sigmoid function to decide whether both samples are the same. In the training stage, the network requires input samples in pairs, either from the same or completely different patterns, given a label of 1 or 0, respectively. Subsequently, through training, the network can measure the similarity of any given pair of samples.

As convolutional neural networks (CNNs) have achieved good results for recognizing HD-sEMG images [38], [39], [40], we chose CNNs to build the first two blocks of both sub-networks for feature learning on EMG images. Fig. 7 further shows the architecture of one subnetwork consisting of three blocks. Block 1 has a convolutional layer with a kernel size of  $3 \times 3$  and 128 filters, followed by a RELU activation function, a batch normalization layer, and a maximum pooling layer with a filter size of  $2 \times 2$ . Block 2 has a convolutional layer with a kernel size of  $2 \times 2$  and 128 filters, followed by a RELU activation function and a batch normalization layer. Block 3 consists of a Flatten layer and a fully connected layer

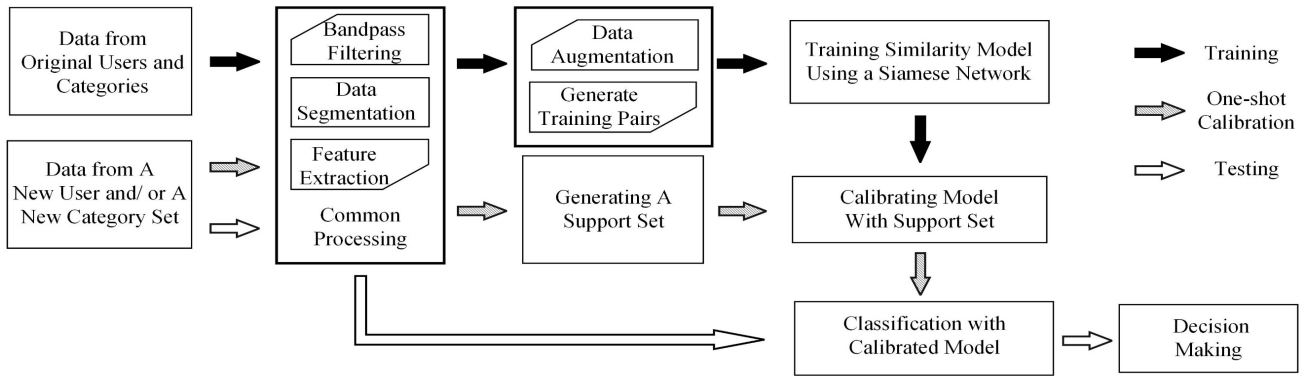


Fig. 5. Flowchart of the proposed method.

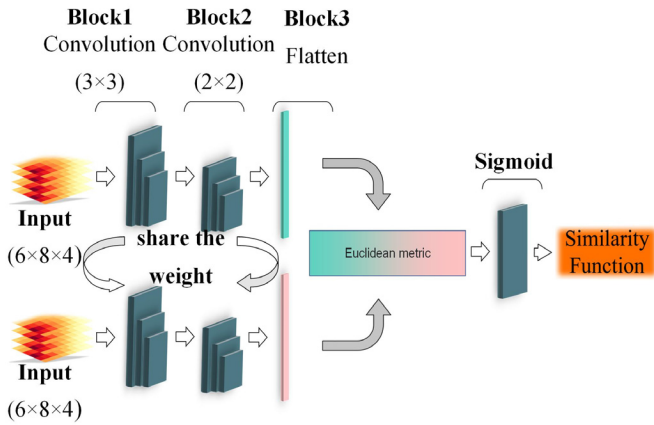


Fig. 6. The architecture of the Siamese neural network involved in the proposed method.

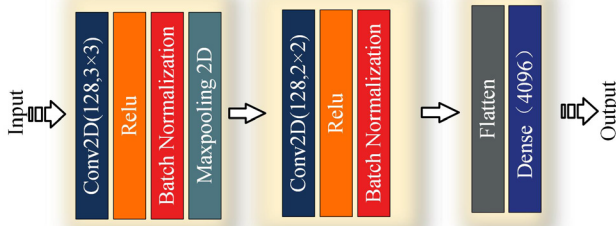


Fig. 7. Subnetwork configuration embedded in Siamese neural networks. Herein, the Conv2D refers to the convolution layer and the two followed notations represent the number and size of the kernel.

with 4096 filters. In the fully connected layer, all weights were initialized from a normal distribution with mean zero and standard deviation of  $10^{-2}$ , and the bias of the fully connected layer was initialized with 0.5 mean and standard deviation of  $10^{-2}$ . In addition, the  $L2$  regularization layer [41] was used after each block to prevent overfitting, with a regularization parameter of  $2 \times 10^{-3}$ .

The ADAM optimizer [42] was applied to train the network with the learning rate set to 0.01. Also, the binary cross-entropy loss was selected as the loss function to optimize the network parameters. The loss function was calculated

as follows:

$$L = y(x_1^{(i)}, x_2^{(i)}) \log P(x_1^{(i)}, x_2^{(i)}) + (1 - y(x_1^{(i)}, x_2^{(i)})) \log(1 - P(x_1^{(i)}, x_2^{(i)})) \quad (1)$$

where  $y(x_1^{(i)}, x_2^{(i)}) = 0$  for the  $i$ -th pair of samples in different patterns and  $y(x_1^{(i)}, x_2^{(i)}) = 1$  for the  $i$ -th pair of samples belonging to the same pattern;  $P(x_1^{(i)}, x_2^{(i)})$  represents the predicted probability distribution. As shown in Fig. 8 (a), the training process represents one iteration per image pair entered into the network, with 200 iterations performed per batch for 100 batches. This means that 20,000 iterations are performed for each training process.

**3) One-Shot Learning Based on the Siamese Neural Network:** The above SNN can determine whether paired input samples are the same. On this basis, a classifier for recognizing multiple patterns can be constructed by a support set containing at least one representative sample from each of all designated gestural categories/patterns. Therefore, such a support set is needed when switching to a new application scenario. The gestural categories can be different from the original categories used for training the SNN, with selection, substitution, and supplementation of certain categories by some new categories. The availability of the support set allows us to build and calibrate a classification model suitable for the new application scenario following the one-shot learning approach.

When the SNN and a support set were determined, the corresponding classifier was also constructed. In the testing phase, as shown in Fig. 8 (b), when an unknown sample (also termed a query category) was input, it was paired with each sample in the support set, and a similarity score of each pair was calculated through the well-trained SNN. The decision was made as the category whose sample pair achieved the greatest similarity score. Referring to the suggestion of Pinheiro and Collobert [43], it can be expressed as:

$$\hat{C} = \arg \max_c \{score_c\} \quad (2)$$

It is worth noting that if the support set was formed by selecting categories directly from the original training data, a classifier supporting classification of the original gestural categories was constructed, which is in line with the basic supervised classification process, without considering any

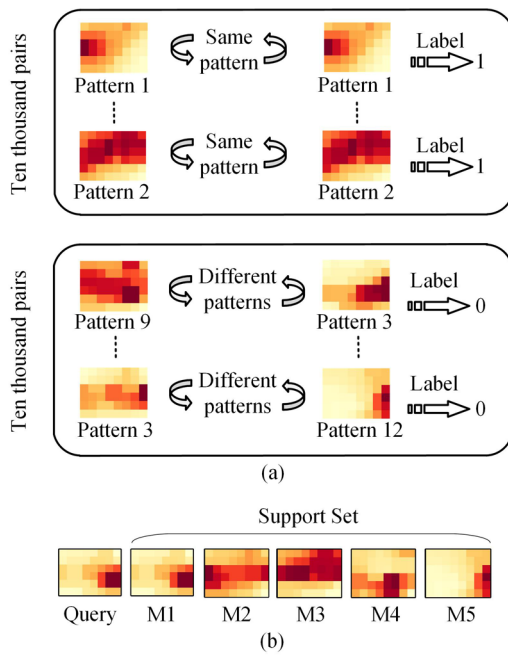


Fig. 8. Illustration of the dataset with well-labeled sample pairs used for training the Siamese neural network (a) and the testing approach with a classifier established under the guidance of a support set. (b).

change of the gestural category set in the new application scenario.

Further, in practice, we may not exclude the possibility of obtaining more than one sample per category from the user during the calibration phase for building the support set. One repetition of a gestural motion usually lasts about 1 s, generating several analysis windows according to the windowing strategy mentioned above. This is equivalent to obtaining a number  $m$  of samples per category to build the support set, i.e., to perform an  $m$ -shot learning approach. In this case, the similarity scores can be calculated for these  $m$  samples of each category separately with the query category, and their averaged value is used as the eventual similarity score for decision making: Any category corresponding to the maximum eventual similarity score was the decision from the classifier.

The proposed method was implemented using the Python language, the Keras framework [44], and the PyQt4 Library. The software ran on a laptop with an Intel Core i5-1035G1 CPU, 16 GB RAM, and NVIDIA MX350 GPU.

#### D. Performance Evaluation

To evaluate the performance better, we designed seven testing scenarios as shown in Fig. 9, including the scenario described in *Online Testing with New Command Set*.

1) The first scenario described a very common MPR testing procedure, termed a “routine” scenario, where both the set of gestural categories and the user for testing remained the same as those in the training dataset. Namely, the Command Set 1 (see Fig. 3) was just considered for testing and training, and the classification approach was implemented in a user-specific manner. As required by the proposed one-shot learning approach, one sample from every gestural category in the training dataset was randomly selected to form the support set that guided the deployment of the classifier. In this scenario,

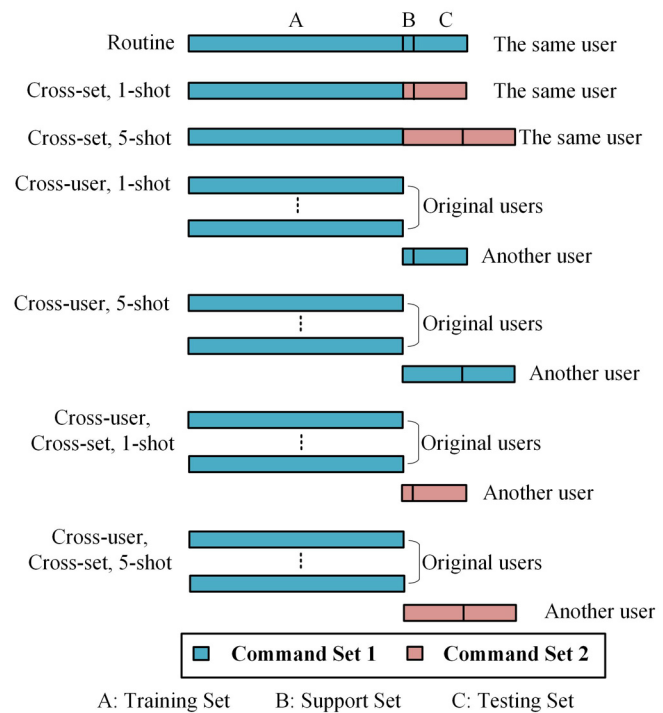


Fig. 9. Schematic plot of the way of testing under five different scenarios. In every scenario, there are three datasets: training dataset, support set, and testing dataset, marked in A, B, and C, respectively. The dataset in a blue bar represents data of gestural categories in Command Set 1, while red bar stands for data from Command Set 2. Data bars in different lines indicate different users.

a five-fold cross-validation strategy was used, where data corresponding to one of five repetitions of the muscle contraction for each category was used for testing, and meanwhile, data of the remaining four repetitions were used for training.

2) The second testing scenario, as described in the online testing with a new command set, was used to test the effectiveness of the proposed method for implementing 1-shot learning in the case of a new command set. It is named “cross-set, 1-shot”.

3) The third test scenario was similar to the previous scenario. When forming the support set, five samples per category were randomly selected, leading to a “cross-set, 5-shot” scenario.

4) In the fourth testing scenario, the user for testing in the scenario was different from any of those providing data to train the SNN, forming a “cross-user, 1-shot” testing scenario. Besides, the set of gestures used for classification was still Command Set 1 and remained consistent. A 7-fold cross-validation scheme was conducted, where data from six subjects were used for training and the data from the remaining subject were tested.

5) The fifth scenario was similar to the fourth scenario. It was consistent with the fourth scenario, except that five samples per category were used to form the support set. It was termed a “cross-user, 5-shot” testing scenario.

6) The sixth scenario is an additional consideration of cross-gesture sets based on the fourth scenario, i.e., the data from five categories in Command Set 2 were used for testing, whereas the data of other categories not in Command Set 2 were used for training and the user for testing in the scenario

was different from any of those providing data to train the SNN, forming a “cross-user, cross-set, 1-shot” testing scenario. Given the cross-user setting, the same 7-fold cross-validation scheme was also conducted, where the testing dataset was formed intentionally from Command Set 2.

7) We intentionally conducted a 5-shot approach to replace the 1-shot design in the sixth scenario. It was termed a “cross-user, cross-set, 5-shot” testing scenario.

For performance comparison, other common MPR methods were also implemented. The conventional MPR method selected KNN as the classifier [45] due to its simple and effective performance. Unlike many traditional methods, with only one sample per category, a KNN classifier can also be trained to work for classification, and the metric idea of the proposed method is similar to that of KNN. [46], [47]. For special testing scenarios, the support set’s data were involved in the training phase for the KNN method. Within state-of-the-art methods relying on transfer learning, the FS-HGR method [30] was adopted because of its successful cross-user/domain applications. In particular, in the above two classification methods, feature extraction and data augmentation methods are consistent with the proposed method. Furthermore, to verify the role of data augmentation, we also implemented both the FS-HGR method and the proposed method without the data augmentation, thus generating two methods termed FWA and PWA (the proposed method without data augmentation), respectively. Other settings of all comparison methods were consistent with those of the proposed method or fine-tuned with optimal performance.

Furthermore, a one-way ANOVA for all five methods was performed for the first scenario. To verify the performance of the proposed method when applied cross-scenarios, a one-way repeated-measures ANOVA was performed on the average accuracy of the remaining six scenarios in the five methods, with both the method (five levels: KNN, FWA, PWA, FS-HGR, Proposed Method) and the scenario (six levels) considered as within-subject factors. The LSD method was employed for post hoc multiple comparisons tests. The significant level was set to 0.05. These statistical analyses were performed using SPSS software (ver. 24.0, SPSS Inc. Chicago, IL, USA).

### III. RESULTS

From the actual decisions of five examples shown in Fig. 10, it can be found that each query sample had a similarity score when it is compared with every element of the support set representing each category. The highest similarity score close to 1 can be observed in the query sample’s category. Deciding by selecting the category achieving the highest similarity score is straightforward. In particular, in the classification process of the query sample at the bottom, both categories give high similarity. This is because both categories have almost visually similar patterns. The classifier struggles to choose the highest similar score as the decision, which is truly correct.

Fig. 11 reports the classification accuracies averaged overall subjects using five different methods under seven testing scenarios. When a routine MPR was conducted (under the routine scenario), it was unsurprisingly found that all methods achieved very high accuracies, close to 100%. When the MPR system was applied across scenarios, however, almost

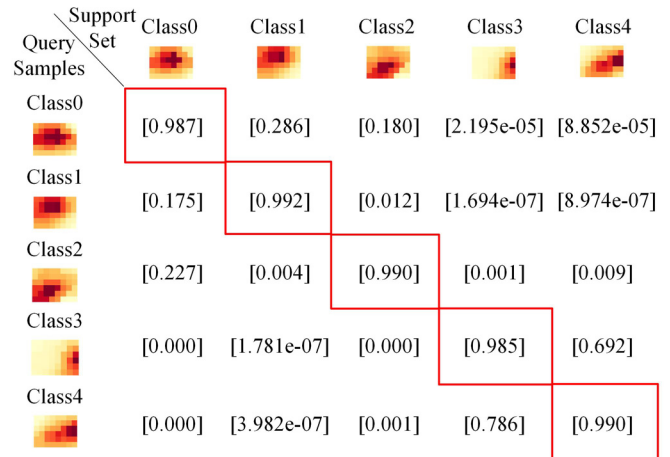


Fig. 10. Five representative examples of the one-shot learning classification. One query sample actually belonging to each of five gestural categories was selected. Each query sample has one for each pattern chosen randomly as an example. The number in the square brackets are the similarity scores, and the command pattern made the decision is in the red box.

all methods had a somewhat compromised performance. The conventional KNN method had its accuracy dramatically drop to  $49.90\% \pm 7.94\%$ . The FS-HGR method yielded an average accuracy of  $71.54\% \pm 8.84\%$ . By contrast, the proposed method had the smallest accuracy decrease among all methods and maintained a relatively high level of accuracy at  $89.73\% \pm 1.99\%$ . In addition, without data augmentation, the FWA method and the PWA method showed a slight performance drop, as compared with the FS-HGR method and the proposed method, respectively. The ANOVA revealed no difference between any two methods under the routine scenario ( $p = 0.811$ ) but a significant difference between these methods under other scenarios involving cross-set and cross-user testing ( $p < 0.001$ ). Specifically, the proposed method significantly outperformed other methods with statistical significance ( $p < 0.05$  for comparing with the PWA method and  $p < 0.001$  for comparing with other methods, respectively).

The confusion matrix in Fig. 12 reports representative results of the FS-HGR (top) and proposed methods (bottom) for one-shot classification of the five new patterns in cross-user and/or cross-set scenarios, respectively. In either case, FS-HGR caused many misclassifications, and these errors were concentrated in both categories G14 and G15. By contrast, the proposed method achieved the most accurate classification of the sample of five new gestures, with only a few sporadic errors present.

Table I reports the computational time costs using the KNN method, the FS-HGR method and the proposed method in a representative cross-scenario application (i.e., cross-set, 1-shot). Before a scenario transfer, a KNN classifier does not need pre-trained, while both FS-HGR and the proposed method need to pre-train a model with a great number of well-labeled training samples, which consumes 28 min and 10 min respectively. However, when switching to a new scenario only the KNN needs to be re-trained/calibrated, the time consumed by the model is 0.057 ms when the data used to calibrate the model has one sample per category. The testing time was expressed as the time consumed by the

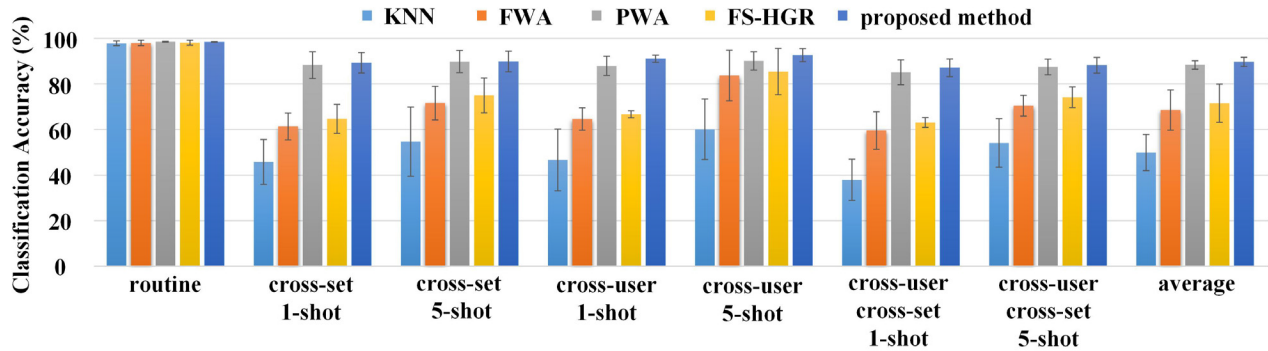


Fig. 11. The mean classification accuracies averaged over all subjects using 5 different methods under when the myoelectric interface was tested on 7 different scenarios, respectively. The condition termed “average” on the right side reports the averaged performance across all scenarios except the routine scenario. Error bars represent standard deviations.

TABLE I  
COMPUTATIONAL TIME COST FOR THE TRAINING  
AND TESTING USING THREE METHODS

|                 | Original Scenario |               | New Scenario       |
|-----------------|-------------------|---------------|--------------------|
|                 | Pre-training      | Re-training   | Testing per sample |
| KNN             | Not necessary     | 0.057 ms      | 0.057 ms           |
| FS-HGR          | 28 min            | Not necessary | 0.351 ms           |
| Proposed method | 10 min            | Not necessary | 0.303 ms           |

model to produce a decision of each sample, where the KNN method achieved the least time consumption of 0.057 ms and the FS-HGR method had the longest one of 0.351 ms. The proposed method required 0.303 ms per testing sample.

#### IV. DISCUSSION

As it evolved into interactive technology, the MPR needs to easily adapt to multiple application scenarios for rapid customization and deployment [48], [49]. This usually requires a large amount of data for re-training or calibration under a new scenario, causing an extra burden that impacts the user’s experience [16], [23], [25]. It is the proposed method that maintains the high performance of the MPR while reducing the re-training burden dramatically using one-shot learning to enhance the flexibility of myoelectric interfaces. The proposed method abandons the traditional classification approach by memorizing each category’s pattern, but it specifically measures similarity between diverse sample pairs. Such learned capability can be well generalized to fast customization of a classifier suitable for any given classification task using the one-shot learning, with the aid of a support set containing just one shot per category. This property allows the classifier built by the proposed method to be fast adapted to various new scenarios.

When no cross-scenario application is involved, as expected, all methods yielded a very high accuracy, close to 100%, and they did not show any significant difference ( $p = 0.695$ ), regardless of whether they were based on the one-shot learning or not. This finding is consistent with most of the previous studies reporting satisfactory performance of MPR under ideal laboratory conditions [10], [24], [27], [30]. When the application scenarios became realistic, i.e., involving new users to instantly manipulate the system with new gestural categories, the classical KNN method encountered a significant performance compromise. This unsurprising finding can be

attributed to the limited generalization of the routine classifier to new users and gestural categories. When one-shot or transfer learning was conducted, including both the FS-HGR method and the proposed method, their performances were not much compromised under the cross-set or cross-user scenarios. Our finding was consistent with previous reports [30], [45] on the application of transfer learning, further confirming the fact that more or less labeled samples from the target domain (new set or new user) help to lead to good adaption and generalization of the classifier.

When testing across different scenarios, the significantly superior performance of the proposed method demonstrates its good generalization and high flexibility in diverse scenarios. Such an advanced property is believed to gain from its capability of assessing similarities or differences between several categories rather than classifying some categories within a predefined and limited range. This mechanism also ensures that good generalization capability can be obtained only with one shot. Our findings (i.e., Fig. 11 and 12) confirmed the significant performance advantage of the proposed method under a one-shot condition compared to the FS-HGR method. The FS-HGR method still had degraded performance under the one-shot condition compared to the 5-shot condition, while the proposed method achieved comparably high performance. Usually, the classifier constructed under the guidance of conventional transfer learning methods is likely to result in an obvious compromise when testing across scenarios once data for calibration is not sufficiently large [1], [4]. These shortcomings can be well compensated with the advance of the proposed method enabling one-shot transfer learning, as demonstrated by the experimental results.

It is worth noting that the user’s re-training burden in the current study mainly consists of the re-collection of the EMG data from the target domain and the computational time consumed by the model re-training or calibration. For both FS-HGR and the proposed method, not only the time to re-collection EMG data is significantly reduced, but also the computational time for model re-training is completely eliminated. Although the KNN method achieved satisfactory recognition accuracy in the “routine” testing scenario as well, 300 samples per class were used to ensure stable performance of the KNN model [46], [50]. However, the time to generate 300 windows for one category would be more than 5 min (given the window increment of 64 ms). In the proposed



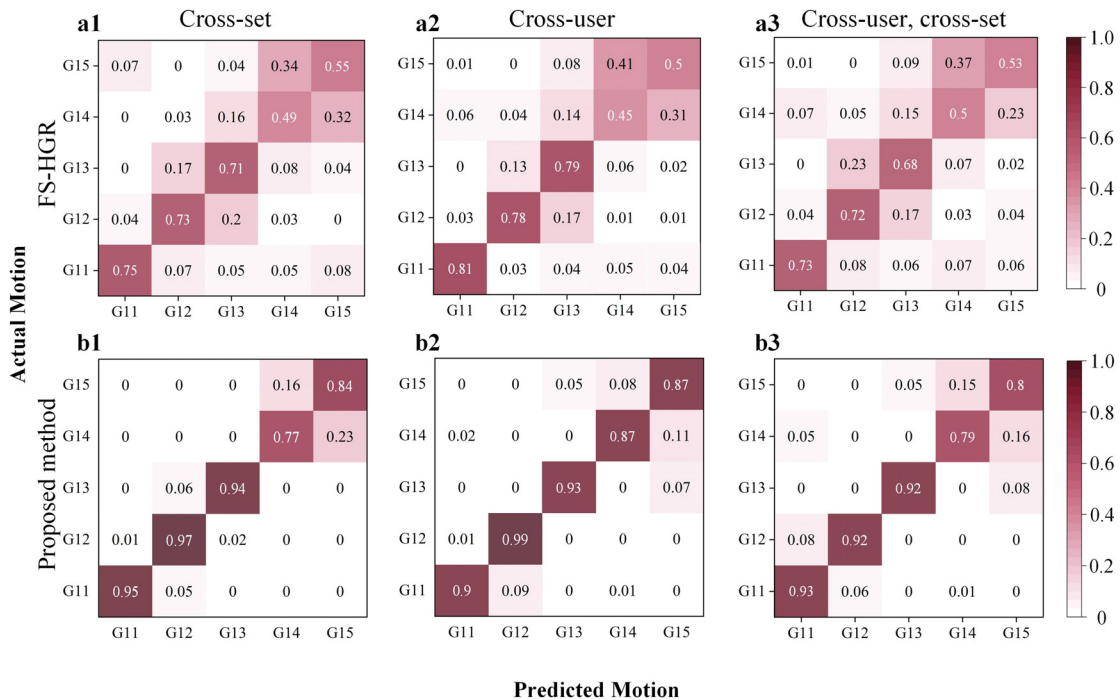


Fig. 12. Confusion matrixes illustrating results of five task patterns (G11-G15) using the FS-HGR method (a) and the proposed method (b), when testing in 1) cross-set, 2) cross-user and 3) both cross-set and cross-user scenario, respectively. The 1-shot condition is consistently applied.

method, by contrast, just 1 shot per category is required to guarantee the model's generalization ability in the target domain, indicating a minimal cost of time for retraining. Under the same 1-shot condition, the FS-HGR method failed to achieve satisfactory performance and had significant accuracy compromise as compared to that under the 5-shot condition (Fig. 11). Moreover, although the model pre-training time is indeed long for the proposed method, such process can be computed in advance from a large amount of offline data and the end-user does not feel the presence of this process. That is, as long as the data re-collection and re-training time is short enough, a good end-user experience can be warranted.

In addition, both the FS-HGR and the proposed method applied a data augmentation approach to improve the diversity of the training dataset, which allowed the network to gain advanced capability of characterizing spatial patterns. Both methods' superior performance confirmed their necessity compared to their non-augmentation versions, i.e., FWA and PWA, respectively. This finding is consistent with previous studies described by Wu et al. [27]. This also falls within the common sense of deep learning that enlarged training data help to learn improved model capability and generalization.

Ideally, the commercialized product of myoelectric control is expected to be a plug-and-play system applicable in any scenario without any re-training or calibration burden. In this regard, our solution using one-shot learning helps to enhance the flexibility of myoelectric interfaces and dramatically reduces the re-training burden to a minimal level. However, it still does not meet the ideal condition where the calibration of the classifier worked in a completely unsupervised manner. This remains to be the major limitation of this study. Furthermore, it is much more interesting to combine

more prior knowledge of skeletal and physiological anatomy to enhance the learned capability of identifying new gestural categories. These abovementioned topics will be important directions for our future efforts.

## V. CONCLUSION

This study presents an MPR method to improve its flexibility in actual applications across different scenarios. The method trains a similarity evaluation model in SNN from a large amount of data paired into image pairs and applies it to new users/new patterns, with the support of one shot per new command set to complete the gestural pattern classification. In applications with switch to different scenarios, including cross-set and cross-user changes, the proposed method achieves up to 90% accuracy, significantly surpassing conventional MPR and common transfer learning methods. It is an effective solution to the high cross-scenario re-training burden in implementing MPR systems.

## REFERENCES

- [1] M. A. Oskoei and H. Hu, "Myoelectric control systems—A survey," *Biomed. Signal Process. Control*, vol. 2, no. 4, pp. 275–294, 2007.
- [2] P. Parker, K. Englehart, and B. Hudgins, "Myoelectric signal processing for control of powered limb prostheses," *J. Electromyogr. Kinesiol.*, vol. 16, no. 6, pp. 541–548, Dec. 2006.
- [3] L. J. Hargrove, K. Englehart, and B. Hudgins, "A comparison of surface and intramuscular myoelectric signal classification," *IEEE Trans. Biomed. Eng.*, vol. 54, no. 5, pp. 847–853, May 2007.
- [4] D. Farina et al., "The extraction of neural information from the surface EMG for the control of upper-limb prostheses: Emerging avenues and challenges," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 22, no. 4, pp. 797–809, Jul. 2014.
- [5] Z. Lu, K.-Y. Tong, X. Zhang, S. Li, and P. Zhou, "Myoelectric pattern recognition for controlling a robotic hand: A feasibility study in stroke," *IEEE Trans. Biomed. Eng.*, vol. 66, no. 2, pp. 365–372, Feb. 2019.

- [6] X. Zhang and P. Zhou, "High-density myoelectric pattern recognition toward improved stroke rehabilitation," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 6, pp. 1649–1657, Jun. 2012.
- [7] J. Kim, E. Kim, S. Park, and J. Kim, "Implementation of a sign language primitive framework using EMG and motion sensors," in *Proc. IEEE 5th Global Conf. Consum. Electron.*, Oct. 2016, pp. 1–2.
- [8] T. Taneichi and M. Toda, "Fighting game skill evaluation method using surface EMG signal," in *Proc. 1st IEEE Global Conf. Consum. Electron.*, Oct. 2012, pp. 106–107.
- [9] X. Zhang, X. Chen, W. Wang, J. Yang, V. Lantz, and K. Wang, "Hand gesture recognition and virtual game control based on 3D accelerometer and EMG sensors," in *Proc. 14th Int. Conf. Intell. Interfaces*, 2009, pp. 401–406.
- [10] U. Côté-Allard et al., "Unsupervised domain adversarial self-calibration for electromyography-based gesture recognition," *IEEE Access*, vol. 8, pp. 177941–177955, 2020.
- [11] E. J. Scheme, B. S. Hudgins, and K. B. Englehart, "Confidence-based rejection for improved pattern recognition myoelectric control," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 6, pp. 1563–1570, Jun. 2013.
- [12] E. Scheme and K. Englehart, "Electromyogram pattern recognition for control of powered upper-limb prostheses: State of the art and challenges for clinical use," *J. Rehabil. Res. Develop.*, vol. 48, no. 6, pp. 1–18, 2011.
- [13] C. M. Light, P. H. Chappell, B. Hudgins, and K. Englehart, "Intelligent multifunction myoelectric control of hand prostheses," *J. Med. Eng. Technol.*, vol. 26, no. 4, pp. 139–146, 2002.
- [14] U. Côté-Allard et al., "A transferable adaptive domain adversarial neural network for virtual reality augmented EMG-based gesture recognition," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 546–555, 2021.
- [15] X. Chen and Z. J. Wang, "Pattern recognition of number gestures based on a wireless surface EMG system," *Biomed. Signal Process. Control*, vol. 8, no. 2, pp. 184–192, Mar. 2013.
- [16] A. Fleming, N. Stafford, S. Huang, X. Hu, D. P. Ferris, and H. Huang, "Myoelectric control of robotic lower limb prostheses: A review of electromyography interfaces, control paradigms, challenges and future directions," *J. Neural Eng.*, vol. 18, no. 4, Aug. 2021, Art. no. 041004.
- [17] N. Jiang and D. Farina, "Myoelectric control of upper limb prosthesis: Current status, challenges and recent advances," *Front. Neuroeng.*, vol. 7, no. 4, pp. 7–9, 2014.
- [18] O. W. Samuel et al., "Intelligent EMG pattern recognition control method for upper-limb multifunctional prostheses: Advances, current challenges, and future prospects," *IEEE Access*, vol. 7, pp. 10150–10165, 2019.
- [19] Y. Zou and L. Cheng, "A transfer learning model for gesture recognition based on the deep features extracted by CNN," *IEEE Trans. Artif. Intell.*, vol. 2, no. 5, pp. 447–458, Oct. 2021.
- [20] F. Zhuang et al., "A comprehensive survey on transfer learning," *Proc. IEEE*, vol. 109, no. 1, pp. 43–76, Jul. 2020.
- [21] K. Zhou, Y. Yang, T. Hospedales, and T. Xiang, "Learning to generate novel domains for domain generalization," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 561–578.
- [22] J. Wang et al., "Generalizing to unseen domains: A survey on domain generalization," *IEEE Trans. Knowl. Data Eng.*, early access, May 26, 2022, doi: [10.1109/TKDE.2022.3178128](https://doi.org/10.1109/TKDE.2022.3178128).
- [23] A. Ameri, M. A. Akhaee, E. Scheme, and K. Englehart, "A deep transfer learning approach to reducing the effect of electrode shift in EMG pattern recognition-based control," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 28, no. 2, pp. 370–379, Feb. 2020.
- [24] R. Hu, X. Chen, X. Zhang, and X. Chen, "Adaptive electrode calibration method based on muscle core activation regions and its application in myoelectric pattern recognition," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 11–20, 2021.
- [25] X. Chen, Y. Li, R. Hu, X. Zhang, and X. Chen, "Hand gesture recognition based on surface electromyography using convolutional neural network with transfer learning method," *IEEE J. Biomed. Health Informat.*, vol. 25, no. 4, pp. 1292–1304, Apr. 2021.
- [26] G. Wilson and D. J. Cook, "A survey of unsupervised deep domain adaptation," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 5, pp. 1–46, Sep. 2020.
- [27] L. Wu, X. Zhang, K. Wang, X. Chen, and X. Chen, "Improved high-density myoelectric pattern recognition control against electrode shift using data augmentation and dilated convolutional neural network," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 28, no. 12, pp. 2637–2646, Dec. 2020.
- [28] S. Kanoga, T. Hoshino, and H. Asoh, "Semi-supervised style transfer mapping-based framework for sEMG-based pattern recognition with 1- or 2-DoF forearm motions," *Biomed. Signal Process. Control*, vol. 68, Jul. 2021, Art. no. 102817.
- [29] W. Guo, M. Wang, X. Sheng, and X. Zhu, "Towards semi-supervised myoelectric finger motion recognition based on spatial motor units activation," *Sci. China Technol. Sci.*, vol. 65, pp. 1232–1242, May 2022.
- [30] E. Rahimian, S. Zabihi, A. Asif, D. Farina, S. F. Atashzar, and A. Mohammadi, "FS-HGR: Few-shot learning for hand gesture recognition via electromyography," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 1004–1015, 2021.
- [31] E. Rahimian, S. Zabihi, A. Asif, S. F. Atashzar, and A. Mohammadi, "Trustworthy adaptation with few-shot learning for hand gesture recognition," in *Proc. IEEE Int. Conf. Auto. Syst. (ICAS)*, Aug. 2021, pp. 1–5.
- [32] S. Chanda, G. V. A. Chakrapani, A. Brun, A. Hast, U. Pal, and D. Doermann, "Face recognition—A one-shot learning perspective," in *Proc. 15th Int. Conf. Signal-Image Technol. Internet-Based Syst. (SITIS)*, Nov. 2019, pp. 113–119.
- [33] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *Proc. ICML Deep Learn. Workshop*, vol. 2, 2015, pp. 1–30.
- [34] S. Pasinetti, M. Lancini, I. Bodini, and F. A. Docchio, "A novel algorithm for EMG signal processing and muscle timing measurement," *IEEE Trans. Instrum. Meas.*, vol. 64, no. 11, pp. 2995–3004, Nov. 2015.
- [35] 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.
- [36] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," 2017, *arXiv:1712.04621*.
- [37] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a 'Siamese' time delay neural network," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 6, 1993, pp. 1–8.
- [38] 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.
- [39] J. Chen, S. Bi, G. Zhang, and G. Cao, "High-density surface EMG-based gesture recognition using a 3D convolutional neural network," *Sensors*, vol. 20, no. 4, p. 1201, Feb. 2020.
- [40] W. Wei, Y. Wong, Y. Du, Y. Hu, M. Kankanalli, and W. Geng, "A multi-stream convolutional neural network for sEMG-based gesture recognition in muscle-computer interface," *Pattern Recognit. Lett.*, vol. 119, pp. 131–138, Mar. 2017.
- [41] C. Cortes, M. Mohri, and A. Rostamizadeh, "L2 regularization for learning kernels," 2012, *arXiv:1205.2653*.
- [42] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
- [43] P. Pinheiro and R. Collobert, "Recurrent convolutional neural networks for scene labeling," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 82–90.
- [44] F. Chollet, "Keras: Deep learning library for Theano and tensorflow," Tech. Rep., 2015, pp. 1–21. [Online]. Available: <https://keras.io/getting-started/faq/#how-should-i-cite-keras>
- [45] G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, "KNN model-based approach in classification," in *Proc. OTM Confederated Int. Conf. 'Move Meaningful Internet Syst.'*, 2003, pp. 986–996.
- [46] M. Z. Al-Faiz, A. A. Ali, and A. H. Miry, "A k-nearest neighbor based algorithm for human arm movements recognition using EMG signals," in *Proc. 1st Int. Conf. Energy, Power Control (EPC-IQ)*, Nov./Dec. 2010, pp. 159–167.
- [47] M. Murugappan, "Electromyogram signal based human emotion classification using KNN and LDA," in *Proc. IEEE Int. Conf. Syst. Eng. Technol.*, Jun. 2011, pp. 106–110.
- [48] A. Asghar, S. Jawaid Khan, F. Azim, C. S. Shakeel, A. Hussain, and I. K. Niazi, "Review on electromyography based intention for upper limb control using pattern recognition for human-machine interaction," *Proc. Inst. Mech. Eng., H, J. Eng. Med.*, vol. 236, no. 5, pp. 628–645, May 2022.
- [49] H. M. Radha and A. K. A. Hassan, "A survey on bio-signal analysis for human-robot interaction," *Int. J. Elect. Comput. Eng.*, vol. 12, no. 6, p. 5998, 2022.
- [50] H. Kucuk, C. Tepe, and I. Eminoglu, "Classification of EMG signals by k-nearest neighbor algorithm and support vector machine methods," in *Proc. 21st Signal Process. Commun. Appl. Conf. (SIU)*, Apr. 2013, pp. 1–4.