Intelligence & Robotics

Research Article

Check for updates

Muscle synergy analysis for gesture recognition based on sEMG images and Shapley value

Xiaohu Ao[1,2,3], Feng Wang[1,2,3], Rennong Wang[4], Jinhua She[4]

[1]School of Automation, China University of Geosciences, Wuhan 430074, Hubei, China.
[2]Hubei Key Laboratory of Advanced Control and Intelligent Automation for Complex Systems, Wuhan 430074, Hubei, China.
[3]Engineering Research Center of Intelligent Technology for Geo-Exploration, Ministry of Education, Wuhan 430074, Hubei, China.
[4]School of Engineering, Tokyo University of Technology, Hachioji, Tokyo 192-0982, Japan.

Correspondence to: Prof. Feng Wang, School of Automation, China University of Geosciences, No.388 Lumo Road, Hongshan District, Wuhan 430074, Hubei, China. E-mail: wangfeng@cug.edu.cn

Abstract

Muscle synergy analysis for gesture recognition is a fundamental research area in human-machine interaction, particularly in fields such as rehabilitation. However, previous methods for analyzing muscle synergy are typically not end-to-end and lack interpretability. Specifically, these methods involve extracting specific features for gesture recognition from surface electromyography (sEMG) signals and then conducting muscle synergy analysis based on those features. Addressing these limitations, we devised an end-to-end framework, namely Shapley-value-based muscle synergy (SVMS), for muscle synergy analysis. Our approach involves converting sEMG signals into grayscale sEMG images using a sliding window. Subsequently, we convert adjacent grayscale images into color images for gesture recognition. We then use the gradient-weighted class activation mapping (Grad-CAM) method to identify significant feature areas for sEMG images during gesture recognition. Grad-CAM generates a heatmap representation of the images, highlighting the regions that the model uses to make its prediction. Finally, we conduct a quantitative analysis of muscle synergy in the specific area obtained by Grad-CAM based on the Shapley value. The experimental results demonstrate the effectiveness of our SVMS method for muscle synergy analysis. Moreover, we are able to achieve a recognition accuracy of 94.26% for twelve gestures while reducing the required electrode channel information from ten to six dimensions and the analysis rounds from about 1000 to nine.

Keywords: Shapley-value-based muscle synergy analysis (SVMS), surface electromyography (sEMG) image, gesture recognition, Grad-CAM

1. INTRODUCTION

The muscle synergy of hand gestures plays a critical role in achieving accurate control of robotic hands and intelligent prosthetic limbs. There are many representative studies in related fields. Hu *et al.* proposed an intelligent stretchable capacitive electronic skin to endow soft body robots with high proprioceptive body awareness. The proposed e-skin can accurately capture various complex three-dimensional (3D) deformations of the entire soft body through multi-position capacitive measurements. The signals from the e-skin can be directly converted into a high-density point cloud that depicts the complete geometry through a transformer-based depth architecture. This high PGR proprioceptive system provides millimeter-scale local and global geometric reconstructions that can assist in solving fundamental soft-body robotics problems such as accurate closed-loop control and digital twin modeling [1]. Park *et al.* described the design and control of a wearable robotic device powered by a pneumatic artificial muscle actuator for ankle-foot rehabilitation. A key feature of the device is its soft structure that provides active assistance without limiting the natural degrees of freedom of the ankle. Four actuated artificial muscles assist in dorsiflexion, plantarflexion, inversion, and valgus. The prototype is also equipped with a variety of embedded sensors for gait pattern analysis [2]. Gesture recognition is typically accomplished through the utilization of surface electromyography (sEMG) signals, which are a result of the electrical activity of superficial muscles and nerves on the skin surface [3]. In the field of human-machine interaction, sEMG signals are highly practical due to their non-invasive nature and ease of manipulation [4].

Gesture recognition is a fundamental component of human-machine interaction and heavily relies on the features of sEMG signals for classification and prediction. Conventional methods of gesture recognition involve manual extraction of time-domain (TD) and frequency-domain features [5,6]. However, sEMG-image-based gesture recognition is a promising new technique that has emerged. Geng *et al.* utilized a convolutional neural network (CNN) with one frame of a high-density sEMG signal as input, achieving a gesture recognition rate of 89.30% [7]. In addition, research has focused on improving preprocessing techniques to enhance the accuracy of gesture recognition, such as the use of a sliding window to capture sEMG signals in the time domain [8].

In the neuromuscular control mechanism, the nerve does not control a muscle alone but recruits multiple muscles on the spinal cord layer to form a muscle synergy, where muscles in the same muscle synergy are activated simultaneously. Muscle synergy is considered to be the smallest unit of motor control in the central nervous system. Various methods have been employed for muscle synergy analysis, including non-negative matrix factorization [9], principal component analysis [10], and independent component analysis [11]. Typically, these methods extract the spatial and temporal components of muscle synergy from sEMG signals and then utilize standard eigenvalues to analyze the correlation between data. However, these approaches necessitate prior knowledge to perform correlation analysis between muscles, thus rendering them non-end-to-end. End-to-end means that the input is the raw data, and the output is the result. The classical machine learning approach is to preprocess raw data into features with human a priori knowledge and then use the features to classify the data. The classification result depends on the goodness of the features, so it takes time to design the features. The traditional method of muscle synergy analysis also requires experts to use various methods to extract the spatio-temporal components of muscle synergy and then use standard feature values to analyze the correlation between the data. Moreover, the acquisition of standard eigenvalues as objects is often a challenging task.

The theory of multiple game interactions is a powerful tool for analyzing the value of interactions between different members [12]. This theory seeks to condense all interactions into a single metric, providing a new method for explaining the underlying prototypical features encoded by neural networks. However, as the number of interacting members increases, the computation required to calculate the marginal contribution of each member increases exponentially. Grad-CAM (gradient-weighted class activation mapping) can be used to address this issue, which is an interpretable method for feature visualization and input unit importance attribution in neural networks [13–15]. While previous studies have not investigated the interpretability of gesture recognition methods based on surface electromyographic signals using this technique, we present the first

results of an interpretability analysis of gesture recognition based on CNN[16]. Our approach visualizes the crucial features of muscles and focuses on muscle conjugation analysis to reduce redundant calculations.

To address the issues mentioned above, this paper provides a Shapley-value-based muscle synergy (SVMS) analysis method that quantitatively analyzes both the synergy among single-channel muscles and muscle groups in different gestural movements[17]. First, the sEMG signal is preprocessed using a TD sliding window, and the resulting signal is converted into a color image to improve gesture recognition accuracy. Next, the Grad-CAM interpretable analysis method is utilized to identify significant feature regions for the sEMG image. This approach not only directly reflects the correspondence between a gesture and electrode importance but also reduces computational expense. Finally, the synergistic effects of forearm muscle groups, upper limb extensors, and upper limb flexors related to gesture recognition are quantitatively analyzed using the Shapley value method. The end-to-end functionality of the SVMS method resolves the issue of muscle correlation analysis. The main contributions of this study are summarized as follows:

1. An end-to-end muscle synergy analysis method is designed to analyze the interaction between muscles from a new perspective. This method facilitates exploration of the correlation between upper extremity flexors, upper extremity extensors, and other muscles.
2. A data extraction method based on a sliding time domain window is designed to improve the prediction accuracy of various gestures in the stage of data preprocessing. The Grad-CAM interpretation method is applied to visualize the importance attribution of the inputs, which intuitively reflects the importance of muscles corresponding to each electrode for different gestures.
3. sEMG electrodes are employed as game members to compute the marginal contribution through interactive game theory. This approach enables the quantitative analysis of the interactions between different muscles.

The rest of this paper is organized as follows: Section 2 explains the related methods involved in SVMS, including data preprocessing methods for sEMG images, model building for a CNN, Grad-CAM, and muscle synergy analysis. Section 3 presents experimental and analysis results. Finally, concluding remarks are presented in Section 4.

2. SVMS METHOD

In this section, we first provide the method of converting sEMG signals into sEMG images, which involves converting sEMG signals into greyscale images and then merging the greyscale images into RGB three-channel color sEMG images. In the next step, we present the model of the CNN for the classification of the input color sEMG images on hand gestures. We then provide the Grad-CAM method to visualize the important features of the input color sEMG images and to gain an intuitive understanding of the important feature areas. Finally, we implement a muscle synergy analysis based on SVMS for the muscle groups corresponding to the different electrodes of the 12 gestures.

With the method designed in the above steps, we can realize the muscle synergy analysis of forearm muscle groups and upper-limb flexor and upper-limb extensor, which are commonly used for gesture recognition. Figure 1 shows the overall framework of the SVMS method.

2.1. sEMG image preprocessing
To convert the raw sEMG signal to a color sEMG image, it is necessary to preprocess the raw signal to obtain the sEMG signal parameter matrix. The parameters of the parameter matrix are mapped to 0-255 to obtain a greyscale image, and then the single-channel greyscale image is merged into a multi-channel color image.

We selected the sEMG signals acquired from sparse electrodes in the Ninapro dataset[18]. The twelve gesture images corresponding to this data subset are shown in Figure 2 below. The raw signal is first band-pass filtered
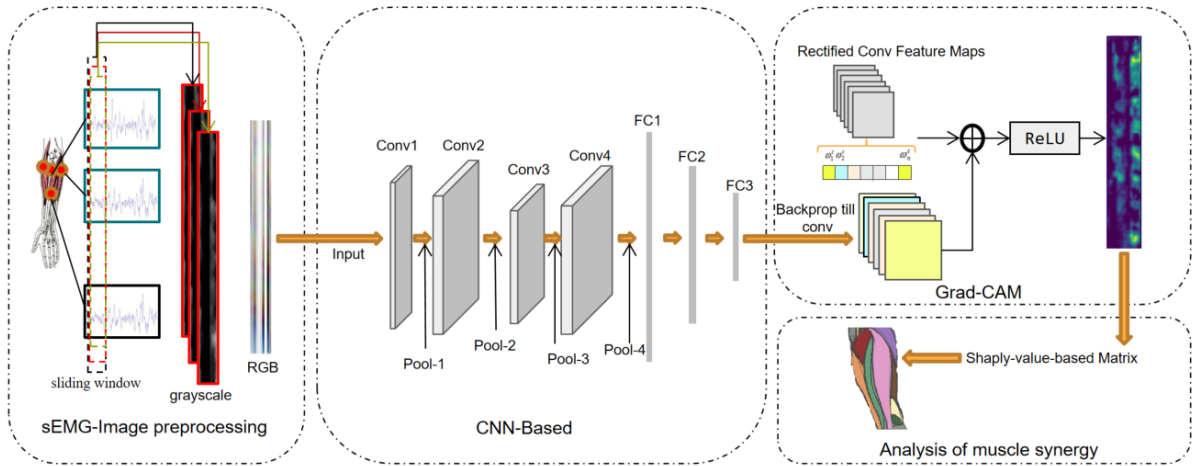
Figure 1. The framework of the Shapley-value-based muscle synergy (SVMS) method.
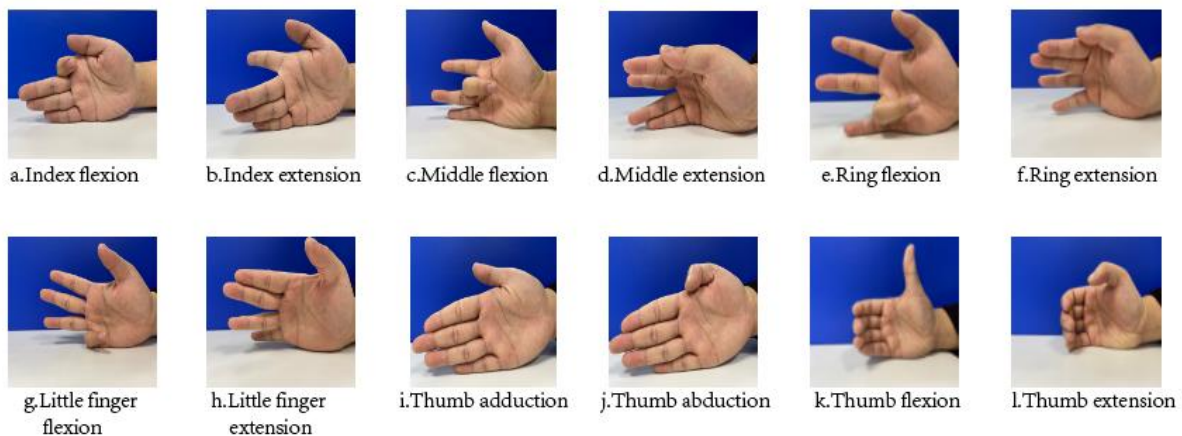

Figure 2. Twelve gestures and corresponding labels.

with a sampling rate of 100 Hz, and a 10-bit A/C conversion is performed[19]. The resulting values are normalized to a range of minus one to plus one, corresponding to a voltage of minus 2.5 mV to plus 2.5 mV. This normalization is based on the maximum and minimum values of all data. The sEMG data from the ten acquisition modules are packaged in an ARM controller[20]. Following the time domain sampling, the sampling duration is defined as one second, and the sEMG signal parameter matrix can be obtained, which is transformed into a sEMG grayscale image. A sliding window is taken; a pre-defined sliding distance is slid in the time domain direction (this sliding distance value is small to ensure that the gap between adjacent grayscale image information is negligible, and data enhancement is also performed to expand the data), and the sliding position is used as the starting point. The sampling duration remains defined as one second, and we can obtain another sEMG grayscale image. By repeating the above steps, the sEMG signal is transformed into a series of sEMG grayscale images. These images are then sorted by gesture labels and time domain. Finally, three adjacent grayscale images are taken for RGB three-channel transformation to obtain the sEMG color images we need, and the above steps are repeated to obtain the sEMG color image set. This image is the input of our network and the object of experimental analysis.

2.2. CNN model for gesture recognition

A CNN is a feed-forward neural network with artificial neurons that respond to a subset of the surrounding units in the coverage area, which is excellent for large image processing[21]. The computational model based on deep CNNs is trained end-to-end, from the original pixel to the final category, without any additional information or manually designed feature extractors. Therefore, this method can effectively fulfill the requirement for experimental validation. A deep learning framework is used to recognize gestures from sEMG images and computationally elucidate patterns in transient sEMG images. We built a network architecture with four convolutional layers and three fully connected layers. This network is the most basic CNN, but the test results are still very good.

2.3. Critical electrode channel selection

Muscle synergy methods are analyzed by manually extracting specific features from sEMG signals. The Grad-CAM interpretable method can be embedded in a CNN, avoiding the step of hand-selecting features and thus obtaining information about muscle activation during the recognition process. In addition to this, Grad-CAM can explain the basis of gesture recognition by the network from a global perspective and get the contribution of features to the gesture recognition task. The electrode channel location where the contributing feature region is located was selected, and this electrode channel was used as the key electrode channel for muscle synergy analysis. Grad-CAM uses the gradient of any target concept that flows into the final convolution layer to generate a coarse localization mapping that highlights important regions of the predicted concept in the image[22]. Given a gesture image as input, we propagate the image through the CNN part of the model and then obtain the raw score for that class by task-specific computation. The gradient of all classes is set to zero except for the desired class, which is set to one. This signal is then back-propagated to the corrected convolutional feature map, and we combine the two to compute the coarse Grad-CAM localization, with the result representing the specific decision the model must look for. Finally, we multiply the heat map with the bootstrap back propagation points to obtain a concept-specific Grad-CAM visualization.

The typical positioning map for the gesture category c, $L^c_{\text{Grad-CAM}} \in \mathbb{R}^{u \times v}$, can be calculated as the following steps. We first compute the fractional gradient y^c of gesture category c before softmax. y^c is the linear categorical logic score of the gesture category c. A^k is a feature map about the convolutional layer, representing each element of the feature map of the kth channel. α^c_k is the neuron importance weights, which are obtained by the global average pooling (GAP) layer. The importance of the kth channel for the gesture category c, α^c_k, is calculated as

$$\alpha^c_k = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A^k_{ij}} \tag{1}$$

This weight α^c_k represents a partial linearization of the underlying deep network from A and obtains the importance of the feature mapping k for the target gesture class c. It is then obtained by performing a weighted combination of forward activation maps through a ReLU, and the ReLU is able to filter irrelevant features. The $L^c_{\text{Grad-CAM}} \in \mathbb{R}^{u \times v}$ is measured as

$$L^c_{\text{Grad-CAM}} = \text{ReLU} \left(\sum_k \alpha^c_k A^k \right) \tag{2}$$

However, class activation mapping (CAM) generates the feature maps of the penultimate layer K, $A^k \in \mathbb{R}^{u \times v}$. These feature maps are then spatially merged using GAP and linearly transformed to generate a score S^c for each gesture class.

$$S^c = \sum_k \omega_k^c \frac{1}{Z} \sum_i \sum_j A_{ij}^k \tag{3}$$

To generate the locality map of the modified image classification constructs, as described above, the order of summation should be exchanged to obtain L_{CAM}^c, and then the score S^c is calculated as

$$S^c = \frac{1}{Z} \sum_i \sum_j \sum_k \omega_k^c A_{ij}^k \tag{4}$$

Grad-CAM can help us understand the process of predicting gestures in CNN models. The high-importance feature region obtained by Grad-CAM allows us to intuitively understand which regions have a greater impact on the network, by which we can invert their corresponding muscles and thus understand which muscles produce more information for that labeled gesture to enable the network to perform the recognition task. On the other hand, this high-importance feature region also helps us to narrow down the channel information interaction and reduces a lot of redundant information for muscle synergy analysis.

2.4. Muscle Synergy Analysis

The total reward value of the gesture recognition process is the output $f(x)$ obtained with all the input information entered into the network minus the output $f(\varnothing)$ obtained without any information entered into the network. Here, '$\varnothing$' denotes an empty input. The total contribution value obtained from the ten electrode channel inputs needs to be fairly distributed to each channel, and to fairly calculate the contribution of each channel individually, we use the Shapley value to consider the effect of the participation or non-participation of that channel in the input on the results for different gesture recognition situations[23]. Therefore, the Shapley value for each input information about the channel is measured as

$$\varnothing_{Shapley}(i) = \sum_{S \subseteq N \setminus \{i\}} \frac{(|N| - |S| - 1)! \, |S|!}{|N|!} \left[f\left(x_{S \cup \{i\}}\right) - f\left(x_S\right) \right] \tag{5}$$

$\varnothing_{Shapley}(i)$ is the Shapley value for the input i. S is the number of electrodes participating in the input, and N is the total number of electrodes. S in formulas is the ten electrode channel information, and $f(x_s)$ denotes the output of the neural network obtained from the input x_S. Previous studies have focused on the interactions between two variables. Given an input I and a total number of participants n, the total reward for all channels is calculated as

$$|f(I) - f(0)| = \sum_{i=1}^n \varphi_i \tag{6}$$

There is a simple definition of interaction. If the input S contains m channels, and the information of these m channels always functions together as input, then these m channels can be considered to form a coalition. The coalition can receive a reward, denoted by ϕ_S. It is usually different from when a single channel acts alone to participate in the game. The additional bonus $\phi_S - \sum_{i \in S} \phi_i$ received by the alliance can be quantified as an interaction. If $\phi_S - \sum_{i \in S} \phi_i > 0$, we consider that the participants in the coalition have a positive influence. On the contrary, it means that there is a negative or antagonistic effect between the group of variables.

However, this equation for measuring interactions can only be used in a single alliance. Their interactions are either purely positive or purely negative. We first discuss the interaction between two single channels

during a certain gesture recognition process. Given two variables i and j, $\phi(i|N)$ and $\phi(j|N)$ denote their Shapley Values, respectively. If the variables i and j always participate in the process together and are always absent together, then they can be considered to form a coalition. The reward received by the coalition is usually different from the sum of the rewards received by the variables i and j when they participate in the whole process alone. This coalition can be considered as a single variable, denoted by S_{ij}. In this way, we can consider the entire process with only $n-1$ channels, including the variable S_{ij} and excluding the variables i and j. The interaction between variables i and j is defined as an additional bonus brought by the coalition S_{ij}. The extra contribution $B(S_{ij})$ made by the cooperation of the variables i and j is calculated as

$$B\left(S_{ij}\right) = \phi\left(S_{ij}|N'\right) - \left[\phi\left(i|N_i\right) + \phi\left(j|N_j\right)\right] \tag{7}$$

The variables i and j cooperate to obtain a high reward, and the interaction is positive if $B\left(S_{ij}\right) > 0$. The interaction between variables i and j leads to a low reward; the interaction is negative if $B\left(S_{ij}\right) < 0$.

We can extend the definition of interaction to multiple variables. We define the channels that always input information together as the set A. N denotes the total number of channels. We consider the channels contained in A as a whole and define it as a single variable and is denoted by $[A]$. The interaction in A is $B([A])$, which is measured as

$$B\left([A]\right) = \phi\left([A]|N_A\right) - \sum_{i \in A} \phi\left(i|N_i\right) \tag{8}$$

By using the SVMS method, we first verify quantitatively whether there is a synergistic effect between forearm muscles. Then, we verify the correlation between a single electrode and a small muscle block for which a single electrode is used. Finally, we define the positively correlated muscle blocks as a coalition and explore the synergy between muscle groups.

3. EXPERIMENT AND ANALYSIS

In this section, we first preprocess the Ninapro dataset to convert it into color images of the sEMG signals. Then, we constructed the CNN model framework and completed the super-parameter setup. We performed Grad-CAM on the image dataset based on the CNN framework to obtain the important feature regions of the network for the input. Finally, we validated the synergy analysis between individual muscles and between muscle groups in these twelve gesture recognition cases using SVMS for the Grad-CAM results.

3.1. sEMG image preprocessing

The Ninapro (Non-Invasive Adaptive Prosthetics) dataset was divided into three sub-databases according to the acquisition procedure and subject characteristics, and we worked on the data from the first database. The first database contains 27 intact subjects (20 males and seven females), 25 of them using the left hand and two using the right hand, and their age was in the range of 28 ± 3.4 years. The acquisition targets sEMG signals from 12 basic finger movements, and the acquisition device contains multiple sensors for recording hand kinetics, kinematics, and corresponding muscle activity. Hand kinetics were measured using the finger force linear sensor FFLS, flexion and extension forces of all fingers, and abduction and adduction of the thumb were detected using strain gauge sensors. Ten MyoBok 13E200-50 electrodes (Otto Bock HealthCare GmbH) provided amplified, band-pass filtered, and root mean square (RMS) corrected versions of the raw sEMG signals corrected version of the original sEMG signal. The amplification gain of the electrodes was set to approximately 14000. The first eight electrodes were placed in an isometric path around the forearm. Electrode
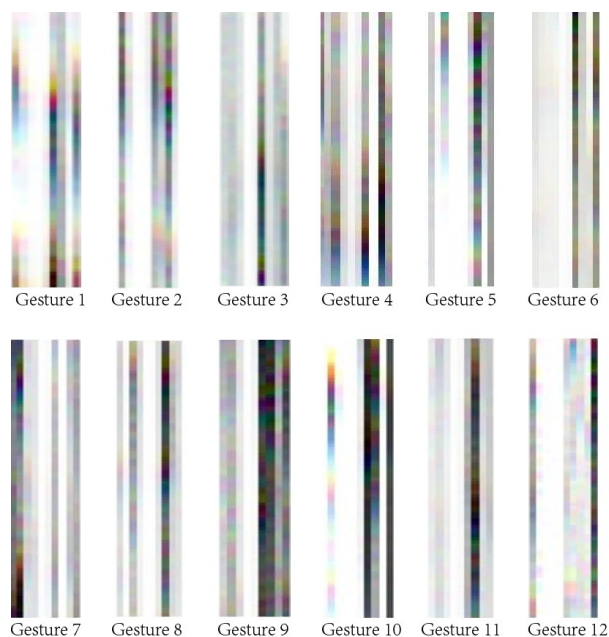
Figure 3. Color images of sEMG signals of twelve hand gestures. sEMG: Surface electromyography.

nine was placed on the upper-limb flexors and electrode ten on the upper-limb extensors. The labels of the dataset were categorized into action modes and resting modes without force by active segment detection. The data was scanned using analysis windows, which can be categorized as overlapping or non-overlapping. In order to obtain more samples, overlapping analysis windows are often used in practical applications. The detected action segments are temporally ordered. Using an overlapping analysis window with a fixed step size, the ten columns of data with the same action labels continue to be partitioned into a series of equally sized arrays under the temporal order. In overlapping analysis windows, the length of the analysis window is an important parameter. In general, the larger the length of the analysis window, the better the action recognition, but the longer the processing time. The response time of a real-time control system should be less than or equal to 300ms; otherwise, it will bring a sense of delay. However, it is crucial to emphasize that the accuracy of the recognition model itself is a prerequisite for ensuring interpretable analysis. At a sEMG signal sampling frequency of 100 Hz, we use an analysis window of size 100*10 and a sliding step of 1 to extract values from the original signal. During this process, we ensure that the gesture labels corresponding to these 100 frames of data are the same. We slide the window in temporal order until we encounter label differences, resulting in a series of arrays of size 100*10. We map the values of these arrays from 0 to 255 and use the fromarray function in the PIL (Python Imaging Library) to convert the 100*10 arrays into grayscale images. After obtaining the grayscale images, the array of three grayscale images adjacent to the same gesture is dimensionally transformed using the swapaxes function. A new 3D array is formed, and this 3D array is transformed to form the sEMG color image. By performing RGB three-channel color image conversion of the three adjacent grayscale images according to the sliding step down-search, we obtain the sEMG color image dataset for each gesture. In the data preprocessing, we used overlapping analysis windows. In the overlapping analysis window, the size of the analysis window and the sliding step size are two important parameters for practical applications. We choose a large window size of 100*10 and a small sliding step size of 1. This choice helps us to obtain a larger amount of data to ensure a better recognition result, which indirectly ensures the accuracy of the subsequent Shapley value calculation. Figure 3 below shows the sEMG signal images for the 12 gestures. Table 1 shows the number of sEMG color images extracted from each gesture in the dataset.

Table 1. Number of color images of sEMG signals for twelve gestures

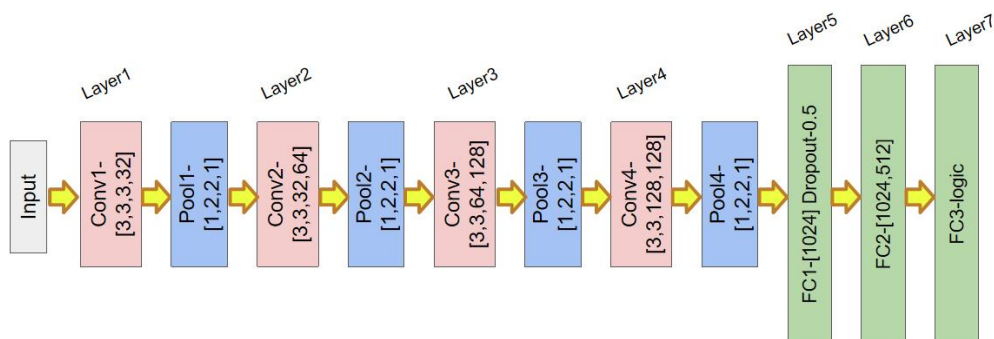| Gesture	a	b	c	d	e	f
Number of pictures	15268	15016	17451	13726	13756	14599
Gesture	g	h	i	j	k	l
Number of pictures	14818	15568	13894	13189	12468	14576

Figure 4. CNN network model constructs and associated hyper-parameters. CNN: Convolutional neural network.

3.2. CNN model framework and hyper-parameter settings

The model we build contains four convolutional layers and three fully connected layers, where the size of the convolutional kernels of the convolutional layers is uniformly set to 3x3. These layers have a stride value of 1 and a padding value of 1 and utilize 32, 64, 128, and 128 convolutional kernels, respectively. Each convolutional layer is followed by a normalization layer and a ReLU activation function. Between each convolutional layer, there is a 2x2 pooling layer. The last three fully connected layers have outputs of 1024, 512, and 12, and then the classification results are output. We chose to build a CNN with only four layers of convolution. This was done to demonstrate the stability of the Shapley value for muscle analysis and highlight its usefulness within a normal network. The loss function is a cross-entropy loss function, and the optimizer is stochastic gradient descent. Figure 4 shows the architecture of the CNN network model. We use these image data as input for a CNN to perform multi-classification recognition tasks and obtain gesture recognition accuracy. At the end of the network, we introduce the Grad-CAM method and perform CAM with gradient weighting on the input images to generate heat maps that show the importance attribution of the network. Based on the information obtained from Grad-CAM, which indicates the input data that the network considers important, we analyze muscle synergy by removing redundant information for the gesture action.

3.3. Experimental results

In this section, we present the experimental results. First, we provide an overview of our network training results, including the training loss, testing loss, training accuracy, and testing accuracy. Then, we showcase the results of the heat intensity map of the CAM, which we obtained by applying the Grad-CAM method. For the high-importance feature regions of the heat intensity map, we identified the target objects for SVMS analysis.

3.3.1 Gesture recognition results

We trained the network for 50 epochs, using 70% of the sEMG color images as the training set and 30% of the sEMG color images as the testing set. We used random seeds, thus ensuring that the data in the test set were not propagated by the neural network during the training process. Then, the network training was performed, and we obtained more satisfactory results, and the prediction accuracy reached 94.26%. Figure 5 illustrates the iterative process of training accuracy and testing accuracy. Figure 6 shows the comparison of the recognition
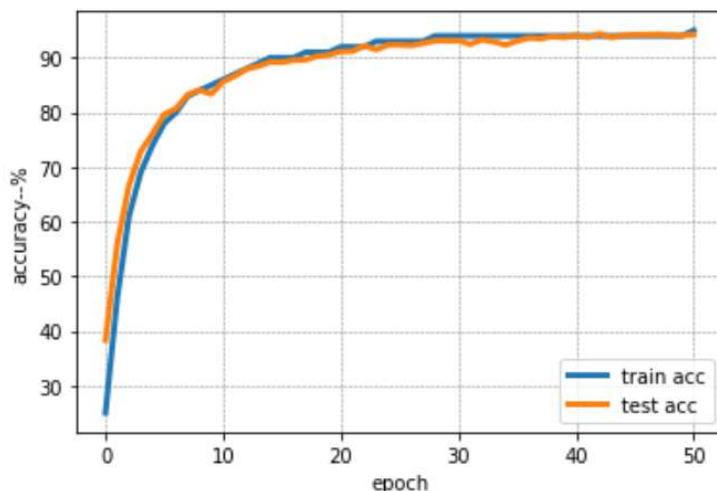
Figure 5. The change of train accuracy and test accuracy for 50 iterations.
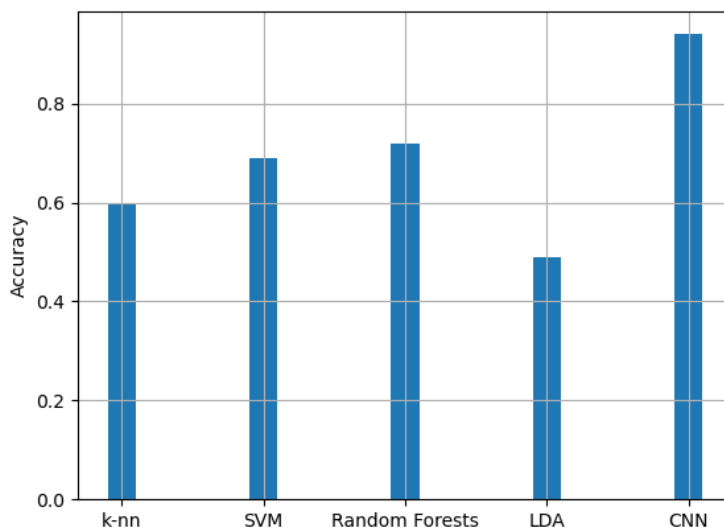

Figure 6. Recognition accuracy of CNN with sliding windows *vs.* traditional machine learning methods. CNN: Convolutional neural network; SVM: shapley-value-based muscle.

accuracy of CNN with sliding windows and traditional machine learning methods.

In comparison to previous machine learning methods, such as k-nn, SVM, Random Forests, and LDA on the Ninapro DB1 dataset, the CNN model achieved higher recognition accuracy in the gesture recognition task.

3.3.2 Grad-CAM results

We let the machine randomize some color images of sEMG signals and apply the Grad-CAM algorithm to calculate the predictions of the network for the regions of interest of the input. The gradient information of the last convolutional layer of the CNN model is first computed, and then the gradient information is weighted and averaged with the output of the convolutional layer to obtain a feature map. Then, the feature map is upsampled to obtain a heat map of the same size as the input image.
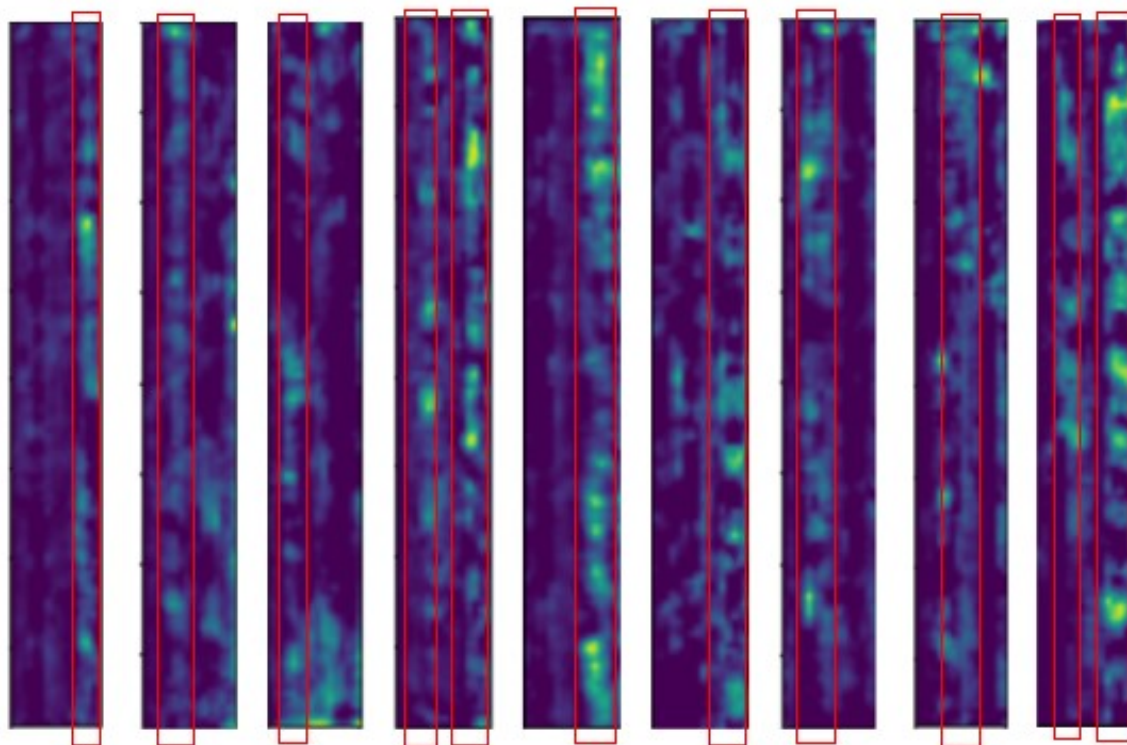
Figure 7. The Grad-CAM plot of the network output. The highlighted region in the plot indicates the region where the network is of high importance for the input features, and the red box in the plot is our label for that region.

Figure 7 shows the results of the Grad-CAM; we find that the regions with attention are presented in a column arrangement, which also intuitively helps us to understand how much attention the network pays to the channels. The black area indicates that the electrode channel information in this section contributes 0 to the gesture recognition task, and not all of the electrode channel acquisition information has a positive effect on the gesture recognition task. Tian *et al.* outlined the significant impact of artificial intelligence technology on the application of sensors [24]. Acquisition devices for sEMG signals also generate redundant information during gesture recognition tasks. Myoelectric acquisition devices have some connection to the engineering task; they could further explore state-of-the-art methods. The red box shows the regions with a high degree of attention. Among these 12 gesture recognitions, we found that the high attention regions of the network for the input are mainly concentrated in columns 4, 6, 7, 8, 9, and 10 by the Grad-CAM graph. Therefore, we mainly discuss the correlation between the forearm muscle groups corresponding to 4, 6, 7, and 8 and the upper-limb flexors and upper-limb extensors corresponding to 9 and 10.

3.4. Muscle Synergy Analysis

The way in which the game interactions are determined determines the Shpeley value calculation process. In order to accurately calculate the contribution value of each electrode channel to the gesture recognition task process, it is necessary to follow a permutation. This ensures that the overall game rounds are guaranteed to cover the interactions between all electrode channels. The combinations function of the itertools toolkit was used to generate all combinations, each of which corresponds to the kinds of game interactions in each round. Input the information of the corresponding electrode channel according to the kind of game interaction in each round. The electrodes appearing in the kind are kept input, and the information in the columns corresponding to the electrode channels not appearing in the kind are all set to zero. The modified image is recognized, and the prediction matrix of that input for the recognition task is taken out before the softmax layer of the network.

This matrix contains the scores for that round for different gesture recognition tasks. Use these scores as the base scores for each gesture. After the combination cases and base scores are calculated for all rounds, the contribution of each channel is obtained using the Shapley value formula. In this section, we first verify that the synergy between different muscles is able to influence the network through a single-channel test. We took the minimum value of the prediction matrix parameters for the whole process as the base value of the membership score, representing the fraction of the contribution that the network considers for that part. We then focused on testing channel electrodes 4, 6, 7, and 8 to explore the synergy between the forearm muscle groups corresponding to these four electrodes and also tested the synergy between the upper-limb flexors and upper-limb extensors corresponding to channel electrodes 9 and 10 in 12 gestures. Finally, we took the inputs of the 4, 6, 7, and 8 electrode channels as one coalition member and the inputs of the 9 and 10 electrode channels as another coalition member and then explored the interactions between these two muscle groups and obtained some results.

3.4.1 Contribution of the input channel

We use the member division in the SVMS method, with ten acquisition channels viewed as ten participants. A picture of gesture 1 is selected from the test set, and we mask columns 2-10 (setting columns 2 to 10 of the picture matrix to 0) and then feed the picture into the network to obtain the prediction matrix. We extract only the value of gesture 1 from the prediction matrix of this image, which is -78.74, as shown in the table below, then mask the first column and columns 3-10, and then feed the image into the network, which is equivalent to using only channel 2 to obtain the result. Again, only the value of -38.11 is taken from the prediction matrix for gesture 1. Repeating the above steps until the ten channels are executed individually as inputs, we get the following prediction matrix for gesture 1.

$$\begin{bmatrix} -78.74 & -38.11 & -167.20 & -27.71 & -62.17 \\ -12.76 & -39.92 & -11.74 & -147.47 & -452.20 \end{bmatrix}$$

Based on the extracted prediction matrix concerning gesture 1 only, we increase its value by an average of 641.82 per person per time as the base score for each member (since the minimum value of the prediction matrix parameter for the whole single-input process is -641.82), then we can obtain the mapped over prediction scores, as shown in the matrix below.

$$\begin{bmatrix} 563.08 & 603.71 & 474.62 & 614.11 & 579.65 \\ 629.06 & 601.90 & 630.08 & 494.35 & 189.62 \end{bmatrix}$$

This prediction score reveals that the maximum value is 630.08, which is the value of channel-8's contribution to gesture 1 in this process. The minimum value is 189.62, which is the contribution of channel 10 to gesture 1. From this, we can conclude that from the single channel input only, channel 8 has the largest predicted contribution value for gesture 1, and channel 10 has the smallest predicted contribution value for gesture 1. We add up the contribution scores of the single channels and compare the total contribution value of 5380.18 for ten channels with inputs turned on simultaneously, and based on equation $\phi_S - \sum_{i \in S} \phi_i$, we obtain the result $\phi_S - \sum_{i \in S} \phi_i > 0$. It verifies that there is a synergistic effect between the muscles. For gesture 2, the prediction matrix scores are

$$\begin{bmatrix} 563.31 & 622.32 & 450.59 & 630.16 & 624.67 \\ 621.66 & 601.37 & 630.39 & 630.79 & 211.05 \end{bmatrix}$$

With the resulting prediction matrix scores, we can see that for gesture 2, the lowest percentage of contribution is made by the information provided by electrode channel 12, and a higher percentage of contribution is made by the information provided by electrode channels 4 to 11. We place the complete calculation results of experiments 2-12 in the Appendix. We can obtain the highest contribution value channel and the lowest contribution value channel for each gesture recognition from the above single-channel prediction score.

3.4.2 Synergy analysis between single channels

We calculate the synergy between muscles and different gestures using game interactions. To achieve this, the six electrodes selected by Grad-CAM are used as game members for 64 rounds of game interactions. We then obtain the marginal contribution of the Shapley value for the six channels. The 64 types of game interactions are obtained by permuting the six participating members. For example, the first round involves only member 1 (channel 1) participating in the recognition task, while the second round involves only member 2 (channel 2) participating, and so on. The 64th round involves all six members (channels) participating in the recognition task together. To calculate the contribution of each channel to the recognition task, we require 64 scores. We use the judgment score of the network before softmax as the reward score for that round for a certain identified gesture. For the results of the above single-channel experiment, we focused on channels 4, 6, 7, 8, 9, and 10 to quantify the correlation between the muscles corresponding to the acquisition electrodes of these channels in this gesture recognition experiment. First, channels 4, 6, 7, and 8 are placed equidistantly around the anterior wall, and the muscle correlations corresponding to these electrodes are quantitatively analyzed. With the additional score matrix, we quantified the effect of the network on the joint action of channel 4 and channel 6. For gestures 8, 9, 10, and 11, the additional interaction of channel 4 and channel 6 is negative. That is, the simultaneous input of channel 4 and channel 6 interferes with the network's judgment of these four gestures. Further, we infer that the network believes that there is a mutual inhibition of the muscles corresponding to channel 4 and channel 6 during the performance of these four gestures, and this inhibition interferes with the recognition accuracy of networks for these gestures. However, none of the additional scores are high, indicating that the muscle interaction corresponding to channel electrode 4 and channel electrode 6 is relatively weak.

Following the method described above, we obtained the one-by-one relationship between channel 4, channel 6, channel 7, channel 8, channel 9, and channel 10 corresponding to the muscles in performing these 12 gestures for recognition. We found that the network perceives different interactions between muscles when performing different gestures and that the joint action between certain muscles and muscles when performing certain recognition facilitates the network's recognition of various gestures. However, this effect is not always positive, and when performing other gestures, certain muscle-muscle interactions inhibit the network's judgments of those gestures. When we performed a synergy analysis between single channels corresponding to muscles, we found that the interactions between channel 6 and channel 8 were mostly biased towards the negative side. The additional fraction matrix for channel 6 and 8 cooperation is shown below.

$$\begin{bmatrix} -27.09 & -14.04 & 52.88 & 52.77 & -2.45 & -82.35 \\ 67.36 & -102.98 & -46.14 & -7.29 & 4.38 & 131.15 \end{bmatrix}$$

The simultaneous input of information from channel 6 and channel 8 negatively affects the network in recognizing gestures 1, 2, 5, 6, 8, 9, and 10. The interaction of information from these two channels has a greater negative impact on the recognition of gesture 6 and gesture 8. But for the recognition of gesture 12, the information from the interaction of these two channels will have a greater positive impact.

3.4.3 Synergy analysis between muscle groups

The six single electrodes were divided into three muscle groups. Specifically, electrodes 4, 6, 7, and 8 were considered as forearm muscle groups, and the muscles corresponding to electrode 9 and electrode 10 were considered as finger extensors and finger flexors, respectively. Discussing the three muscle groups, the number of gaming rounds just needs eight rounds, and calculating eight rounds of gaming interaction, 12 Shapley value arrays are obtained; each array includes the marginal contribution of the corresponding electrode channels of the three muscle groups for the gesture, and the contribution score represents the specific contribution of the muscle group information for this recognition task. The synergistic relationship of different muscle groups for different gestures can then be obtained. In terms of interpreting the results, we aim to address the impact of reducing electrode information and analysis rounds on the overall performance and efficiency of our method. To do so, we used the Grad-CAM method to cope with the large amount of data and high time complexity of the game interaction. The Grad-CAM method reduced the redundant electrode information by locking the ten electrode channels into six electrode channels, resulting in a 40% reduction of the raw data volume. This reduction not only directly reduced the training time of the network but also eliminated the impact of redundant information on the network performance. In addition, the marginal contribution of the Shapley value to the calculation of the game interaction between electrodes is also significant. In previous research, 1024 rounds were required to obtain the synergistic relationship between different muscle groups for different gestures. However, our method reduces the number of electrodes to 6 and considers the muscles corresponding to electrodes 4, 6, 7, and 8 placed equidistantly around the forearm as forearm muscle groups, and the muscles corresponding to electrodes 9 and 10 as extensor and flexor muscles. Focusing on these three muscle groups, the number of game interactions is reduced to only eight rounds to obtain the synergistic relationship between different muscle groups for different gestures. Extend the definition of interaction to multiple variables for muscle groups. One set A is a subset belonging to N. If the variables in A always participate together in some process, then the variables in A form a union, which is considered as a single variable and is denoted by $[A]$. The interaction in A is measured by

$$B\left([A]\right) = \phi\left([A]\,|N_A\right) - \sum_{i \in A} \phi\left(i|N_i\right) \tag{9}$$

When these four channels of information are input together, there is a facilitative effect for the recognition of some gestures and a suppressive effect for the recognition of others. However, the value of the extra score is not high, which also proves the existence of positive and negative effects that cancel each other out. The muscles corresponding to the four channel electrodes are actually close together, so we infer that there are different synergies between the muscles of the small muscle groups when doing different gestures, yet this synergy does not always allow the network to make better recognition results. We explored the synergistic relationship between the forearm muscle groups corresponding to channels 4, 5, 7, and 8 and the upper extremity flexors and upper extremity extensors corresponding to channels 9 and 10. We consider channels 4, 5, 7, and 8 as an alliance and treat that alliance as a member. We then consider 9 and 10 as one member. Using that total contribution score, subtracting the total contribution scores entered simultaneously for channels 4, 5, 7, and 8, and subtracting the total contribution scores entered simultaneously for channels 9 and 10, we obtain the matrix of additional scores for the cooperation of these two members.

$$\begin{bmatrix} 123.38 & 210.06 & 94.60 & 38.69 & 91.17 & -129.36 \\ 158.69 & 206.31 & 6.56 & 9.56 & 119.79 & 41.42 \end{bmatrix}$$

The interaction of these two distant muscle groups could positively contribute to the network's recognition in most cases and only had a greater negative effect on the recognition of gesture 6. Therefore, for the recognition of gesture 6, the network considered the interaction generated by these two muscle groups as negative.
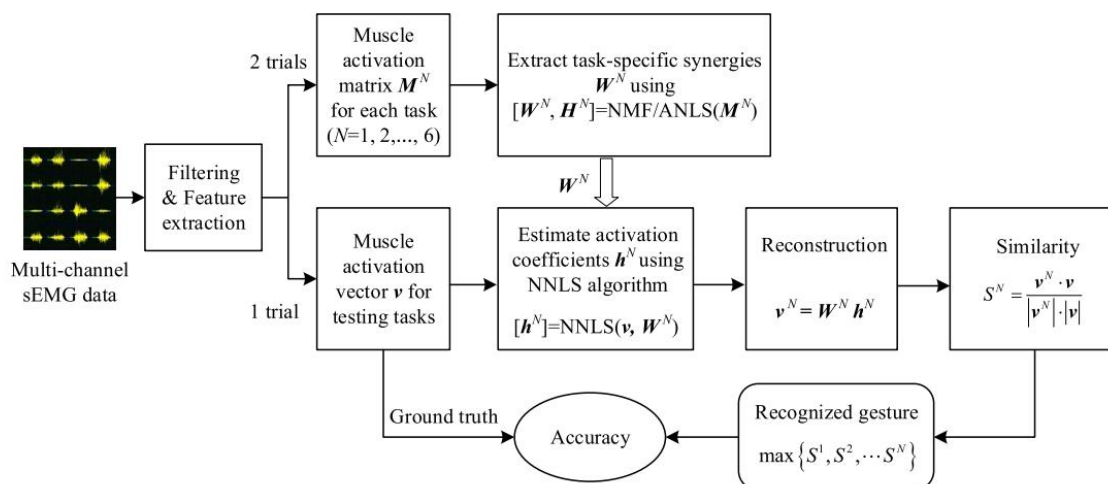
Figure 8. Muscle synergy analysis using non-negative matrix decomposition [25].

3.4.4 Comparison with previous work

In this section, we compare the SVMS method with the classical muscle synergy analysis method. Teng *et al*. utilized synergy for gesture recognition methods with integrity [25]. In their approach, the first step is to model muscle synergy analysis, where high-dimensional MAP (muscle activation patterns) can be represented as linear combinations of low-dimensional muscle synergies activated by corresponding activation coefficients. In the second step, MAP approximations need to be established. sEMG data are extracted through analysis windows, and features are extracted from each analysis window to form a feature matrix for each gesture. This matrix is known as the MAP matrix. Typical TD features, i.e., RMS and mean absolute value (MAV), were used to generate the MAP matrix. In the third step, muscle synergies were extracted and estimated from the MAP matrix using non-negative matrix decomposition, but the number of synergies needed to be determined manually. The non-negative matrix decomposition algorithm generates multiple solutions, and the set of muscle synergies that minimizes the error needs to be extracted for data reconstruction. In addition, the coefficient of determination ($R2$) is needed as a criterion. The reconstruction superiority of the raw sEMG signal data is measured by the estimated muscle synergies and the corresponding activation coefficients.

For feature selection, the SVMS method is performed based on CNNs, avoiding the step of manual feature extraction. For the analysis window, the SVMS method converts the signal into an image for processing and utilizes an interpretable method to filter the information that is important for the recognition task, avoiding the establishment and processing of the feature matrix. In synergy analysis, the SVMS method utilizes the game interaction theory to establish the interaction between information. The manual determination of the number of synergies is avoided. Furthermore, it does not utilize non-negative matrix decomposition to estimate the muscle synergy from the MAP matrix. There is no need to consider the problem that the non-negative matrix decomposition algorithm generates multiple solutions.

4. CONCLUSIONS

In this paper, we introduce a novel and effective method for muscle synergy analysis, named SVMS, which is both interpretable and has end-to-end functionality. The SVMS method employs a time-domain sliding window for data acquisition, achieving a high gesture recognition accuracy of 94.26%. We analyze the synergy between the muscles associated with twelve different gestures and demonstrate the effectiveness of our method. The advantage of the SVMS method is that it explores the correlation between muscles involved in

distinct movements without requiring prior knowledge to define standard eigenvalues, as needed by previous muscle synergy analysis methods. Our method offers a promising solution for muscle synergy analysis in human-machine interaction, with potential applications in fields such as rehabilitation and sports training. In addition, this method can be applied not only to analyze forearm muscle interactions but also in lower limb rehabilitation analysis, gait analysis, human movement recognition, and many other areas. As an example, we have introduced our method for muscle synergy analysis in the context of lower limb rehabilitation. Specifically, it is capable of outputting correlation analysis on lower limb muscles while recognizing the patient's movement intention. This enables us to evaluate the patient's lower limb rehabilitation progress.

In the future, we plan to explore the levels of intra-variability (within an individual) and inter-variability (between individuals), and we plan to conduct an analysis of the effects of various sizes of sEMG images to assess the correlation of synergistic relationships for similar movements. In addition, we aim to incorporate muscle forces and joint angles to analyze changes in muscle synergy during continuous movements. For graphs, each data sample in the graph will have edges associated with other real data samples in the graph, and it is this information that we use to capture the inter-dependencies between muscles. Inspired by Liu *et al*.'s Graph structure learning based on evolutionary computation[26], graph neural networks are introduced to explore the correlation between the data sample points in the sEMG image. Regarding the multi-sensor data fusion model[27], the first step is to determine the data type that will be used. We believe that the sEMG physiological signal is a better choice. When using computer vision for gesture recognition, occlusion can interfere with the recognition task. However, if our method is fused, the sEMG physiological signal can also be used as a feature needed for the network, which may assist in addressing this problem. The design and development of sEMG signal acquisition devices in accordance with practical engineering applications are also worthy of in-depth investigation with respect to the large amount of redundant information derived from the interpretable new analyses. In addition, the large amount of redundant information derived from the new analyses can be interpreted. The design and development of sEMG signal acquisition devices and control of intelligent prosthetics in accordance with practical engineering applications are also worthy of in-depth investigation.

DECLARATIONS

an informed consent form. The Ninapro acquisition experiment was conducted according to the principles expressed in the Declaration of Helsinki (www.wma.net/en/20activities/10ethics/10helsinki) and received approval from the Ethics Commission of the Canton Valais (Switzerland), ensuring the safety of the involved hardware and adherence to the WMA (World Medical Association).

Consent for publication
Not applicable.

Copyright
© The Author(s) 2023.

REFERENCES

1. Hu DL, Giorgio-Serchi F, Zhang SM, Yang YJ. Stretchable e-skin and transformer enable high-resolution morphological reconstruction for soft robots. *Nat Mach Intell* 2023;5:261-72. DOI
2. Park YL, Chen BR, Pérez-Arancibia NO, et al. Design and control of a bio-inspired soft wearable robotic device for ankle–foot rehabilitation. *Bioinspir Biomim* 2014;9:016007. DOI
3. Ding QC, Xiong AB, Zhao XG, Han JD. A review on researches and applications of sEMG-based motion intent recognition methods. *Acta Automatica Sinica* 2016;42:13-25. DOI
4. He JY, Jiang N. Biometric from surface electromyogram (sEMG): feasibility of user verification and identification based on gesture recognition. *Front Bioeng Biotechnol* 2020;8:58. DOI
5. Bourke AK, Brien JV, Lyons GM. Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm. *Gait Posture* 2007;26:194-9. DOI
6. Bretzner L, Laptev I, Lindeberg T. Hand gesture recognition using multi-scale colour features hierarchical models and particle filtering. *Proceedings of Fifth IEEE International Conference on Automatic Face Gesture Recognition* 2002;423-8. DOI
7. Geng WD, Jin WG, et al. Gesture recognition by instantaneous surface EMG images. *Sci Rep* 2016;6:36571. DOI
8. Gijsberts A, Atzori M, Castellini C, Muller H, Caputo B. The movement error rate for evaluation of machine learning methods for sEMG-based hand movement classification. *IEEE Trans Neural Syst Rehabil Eng* 2014;22:735-44. DOI
9. Tang L, Li F, Cao S, Zhang X, Wu D, Chen X. Muscle synergy analysis in children with cerebral palsy. *J Neural Eng* 2015;12: 046017. DOI
10. Hargrove LJ, Li G, Englehart KB, Hudgins BS. Principal components analysis preprocessing for improved classification accuracies in pattern-recognition-based myoelectric control. *IEEE Trans Biomed Eng* 2009;56:1407-14. DOI
11. Morris CM, O'Brien KK, Gibson AM, Hardy JA, Singleton AB. Polymorphism in the human DJ-1 gene is not associated with sporadic dementia with Lewy bodies or Parkinson's disease. *Neurosci Lett* 2003;352:151-3. DOI
12. Zhang H, Xie Y, Zheng L, Zhang D, Zhang Q. Interpreting multivariate shapley interactions in dnns. *AAAI* 2021;35:10877-86. DOI
13. Zeiler MD, Fergus R. Visualizing and understanding convolutional networks. *The 13th European Conference on Computer Vision (ECCV)* 2014;818-33. DOI
14. Ribeiro MT, Singh S, Guestrin C. "why should i trust you?" explaining the predictions of any classifier. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations* 2016;97–101. DOI
15. Selvaraju R R, Cogswell M, Das A. Grad-cam:visual explanations from deep networks via gradient-based localization. *2017 IEEE International Conference on Computer Vision (ICCV)* 2017;618-6. DOI
16. Su C, Yu G, Wang J, Yan Z, Cui L. A review of causality-based fairness machine learning. *Intell Robot* 2022;2:244-74. DOI
17. Atzori M, Gijsberts A, Kuzborskij I, et al. Characterization of a benchmark database for myoelectric movement classification. *IEEE Trans Neural Syst Rehabil Eng* 2015;23:73-83. DOI
18. Atzori M, Gijsberts A, Castellini C, et al. Electromyography data for non-invasive naturally-controlled robotic hand prostheses. *Sci Data* 2014;1:140053. DOI
19. Kuzborskij I, Gijsberts A, Caputo B. On the challenge of classifying 52 hand movements from surface electromyography. *Annu Int Conf IEEE Eng Med Biol Soc* 2012;2012:4931–7. DOI
20. Tenore FVG, Ramos A, Fahmy A, Acharya S, Etienne-Cummings R, Thakor NV. Decoding of individuated fifinger movements using surface electromyography. *IEEE Trans Biomed Eng* 2009;56:1427-34. DOI
21. LeCun Y, Jackel LD, Boser B, et al. Handwritten digit recognition: applications of neural network chips and automatic learning. *IEEE Commun Mag* 1989;27:41-6. DOI
22. Selvaraju R, Abhishek D, Vedantam D, Cogswell M, Parikh D, Batra D. Grad-CAM: why did you say that? *NIPS 2016 Workshop on Interpretable Machine Learning in Complex Systems* 2016. DOI
23. Shapley LS. A value for n-person games. *ontributions to the Theory of Games* 1953;28:307-318. DOI
24. Tian CP, Xu ZY, Wang LK, Liu YJ. Arc fault detection using artificial intelligence: challenges and benefits. *Math Biosci Eng.* 2023;20:12404-32. DOI
25. Teng ZC, Xu GH, Liang RH, Li M, Zhang SC. Evaluation of synergy-based hand gesture recognition method against force variation for

robust myoelectric control. *IEEE Trans Neural Syst Rehabil Eng* 2021;29:2345-54. DOI

26. Liu ZW, Dong Y, Wang YJ, Lu MJ, Li RR. EGNN: graph structure learning based on evolutionary computation helps more in graph neural networks. *Applied Soft Computing* 2023;135:110040. DOI

27. Wen Q, Salih EO, Zhijun L, Aldo M, Rong S. Multi-sensor guided hand gesture recognition for a teleoperated robot using a recurrent neural network. *IEEE Robot Autom Lett* 2021;6:6039-45. DOI

APPENDIX

Performing a second round of process experiments, we select the sEMG signal images of gesture 2 for recognition and continue to input the information from each of the ten channels until all 12 gesture pictures we selected from the test set are experimented with by the above process. Consequently, we can obtain the prediction matrix of each channel for each gesture at a single channel input.

$$
\begin{bmatrix}
-78.51 & -19.50 & -191.23 & -11.66 & -17.15 & -20.16 & -40.45 & -11.43 & -11.03 & -430.77 \\
-91.37 & -16.43 & -119.60 & -23.52 & -34.99 & -7.18 & -15.09 & -62.18 & -44.95 & -288.01 \\
-23.40 & -19.43 & -25.50 & -27.21 & -37.94 & -12.22 & -32.39 & -102.60 & -49.84 & -243.33 \\
-73.72 & -35.78 & -217.70 & -38.36 & -22.00 & -16.16 & -26.53 & -4.95 & -116.97 & -74.30 \\
-96.91 & -15.30 & -258.58 & -27.08 & -33.50 & -12.12 & -7.01 & -23.02 & -98.78 & -41.82 \\
-90.20 & -18.63 & -170.133 & -13.20 & -17.22 & -20.73 & -61.80 & -90.17 & -22.50 & -225.76 \\
-80.86 & -33.91 & -64.15 & -16.84 & -8.64 & -15.52 & -36.74 & -6.90 & -42.39 & -104.97 \\
-122.00 & -7.97 & -23.65 & -28.18 & -27.10 & -16.74 & -7.39 & -5.65 & -4.40 & -6.22 \\
-158.97 & -133.55 & -213.51 & -13.60 & -26.89 & -6.29 & -4.27 & -2.86 & -52.04 & -11.51 \\
-132.75 & -20.86 & -131.37 & -21.03 & -65.93 & -31.46 & -55.55 & -4.05 & -60.07 & -175.16 \\
-28.10 & -37.01 & -59.62 & -29.08 & -23.43 & -30.33 & -46.56 & -116.75 & -235.59 & -9.52
\end{bmatrix}
$$

The following is the individual channel input prediction scoring matrix for rounds 2 through 11, with rows representing gesture categories and columns representing channel categories.

$$
\begin{bmatrix}
563.31 & 622.32 & 450.59 & 630.16 & 624.67 & 621.66 & 601.37 & 630.39 & 630.79 & 211.05 \\
550.45 & 625.39 & 522.22 & 618.30 & 606.83 & 634.64 & 626.73 & 579.64 & 596.87 & 353.81 \\
618.42 & 622.39 & 616.32 & 614.61 & 603.88 & 629.60 & 609.43 & 539.22 & 591.98 & 398.49 \\
568.10 & 606.04 & 424.12 & 603.46 & 619.82 & 625.66 & 615.29 & 636.87 & 524.85 & 567.52 \\
544.91 & 626.52 & 383.24 & 614.74 & 608.32 & 629.70 & 634.81 & 618.80 & 543.04 & 600.00 \\
551.62 & 623.19 & 471.69 & 628.62 & 624.60 & 621.09 & 580.02 & 551.65 & 619.32 & 416.06 \\
560.96 & 607.91 & 577.67 & 624.98 & 633.18 & 626.30 & 605.08 & 634.92 & 599.43 & 536.85 \\
519.82 & 633.85 & 618.17 & 613.64 & 614.72 & 625.08 & 634.43 & 636.17 & 637.42 & 635.60 \\
482.85 & 628.27 & 428.31 & 628.22 & 615.23 & 635.53 & 637.55 & 638.96 & 589.78 & 630.31 \\
509.07 & 620.96 & 510.45 & 620.79 & 575.89 & 610.36 & 586.27 & 637.77 & 581.75 & 466.66 \\
613.72 & 604.81 & 582.20 & 612.74 & 618.39 & 611.49 & 595.26 & 525.07 & 406.23 & 632.30
\end{bmatrix}
$$

The matrix of additional scores for channel four and channel six is shown below.

$$
\begin{bmatrix}
35.14 & 19.52 & 13.30 & 10.27 & 44.62 & 20.98 \\
26.77 & -33.47 & -30.13 & -21.46 & -7.95 & 29.95
\end{bmatrix}
$$

The cooperative extra score matrix for the input information of channels 9 and 10 is shown below.

$$
\begin{bmatrix}
501.33 & 243.07 & 266.16 & 254.71 & 148.65 & 36.60 \\
168.36 & -12.20 & -38.68 & 53.34 & 126.31 & 221.55
\end{bmatrix}
$$

The simultaneous input of information from channel 9 and channel 10 only causes a small negative impact on the network when recognizing gestures 8 and 9. However, the additional contribution of these two channels is high, meaning that the interaction of information from these two channels has a large positive impact, especially for the recognition of gestures 1, 2, 3, 4, 5, 7, 11, and 12 with a nice boost. We verify the effect of the network on the input of these four channels together. The cooperative extra score matrix for the input information of channels 4, 6, 7, and 8 is shown below.

$$\begin{bmatrix} 47.57 & 64.87 & 70.58 & 113.25 & -7.20 & 41.87 \\ 79.13 & -31.46 & -9.51 & -0.43 & 89.26 & 174.96 \end{bmatrix}$$