Information Sciences 672 (2024) 120667

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Explainable deep learning for sEMG-based similar gesture recognition: A Shapley-value-based solution

Feng Wang [a,b,c], Xiaohu Ao [a,b,c], Min Wu [a,b,c], Seiichi Kawata [a,b,c], Jinhua She [d,*]

[a] School of Automation, China University of Geosciences, Wuhan 430074, China
[b] Hubei Key Laboratory of Advanced Control and Intelligent Automation for Complex Systems, Wuhan 430074, China
[c] Engineering Research Center of Intelligent Technology for Geo-Exploration, Ministry of Education, Wuhan 430074, China
[d] School of Engineering, Tokyo University of Technology, Hachioji, Tokyo 192-0982, Japan

ARTICLE INFO

Keywords:
Explainable artificial intelligence
Deep learning
Similar gesture recognition
Shapley value

ABSTRACT

Surface electromyography (sEMG) based gesture recognition shows promise in enhancing human-robot interaction. However, accurately recognizing similar gestures is a challenging task, and the underlying mechanisms of gesture recognition are not well understood. To address these issues, we developed a new solution called the Shapley-value-based similar gesture recognition (SV-SGR) method. Our solution combines deep learning and game theory to achieve high recognition accuracy and interpretability. First, we devised a data preprocessing method that converts sEMG signals into sEMG color images, which can be more effectively utilized by deep learning techniques. Next, we established a deep-neural-network-based model for gesture recognition using the processed sEMG color images. Then, we designed a global explanation approach based on Shapley values to quantify the contribution of each channel to recognizing similar gestures. Finally, we carried out an explanation analysis, which provides feedback on the recognition model to enhance the precision of gesture recognition. Extensive comparisons and interpretable analyses have been conducted on real-world datasets, and the results demonstrate that the SV-SGR method outperforms other baselines under various experimental conditions. The interpretable analysis method based on Shapley values effectively enhances the performance of recognizing similar gestures and provides valuable insights into the decision-making process of recognition models.

1. Introduction

Advancements in the fields of artificial intelligence and hardware equipment have fueled a growing interest in human-robot interaction [36]. Gesture recognition plays an important role in facilitating this interaction, particularly in rehabilitation and assistive devices such as robotic prostheses and hand exoskeletons [32]. Among the signals utilized in gesture recognition, surface electromyography (sEMG) signals hold significant importance [27]. These signals are captured by sensors placed on the skin above the muscles responsible for hand gestures and subsequently processed and analyzed using various signal processing techniques and artificial intelligence algorithms to identify the specific muscle activation patterns associated with each gesture. One of the great challenges in sEMG-based gesture recognition is the discrimination of similar hand gestures, encompassing grasps, functional movements, and

* Corresponding author.
E-mail address: she@stf.teu.ac.jp (J. She).

https://doi.org/10.1016/j.ins.2024.120667
Received 2 October 2023; Received in revised form 28 April 2024; Accepted 28 April 2024

Available online 3 May 2024
0020-0255/© 2024 Elsevier Inc. All rights reserved.

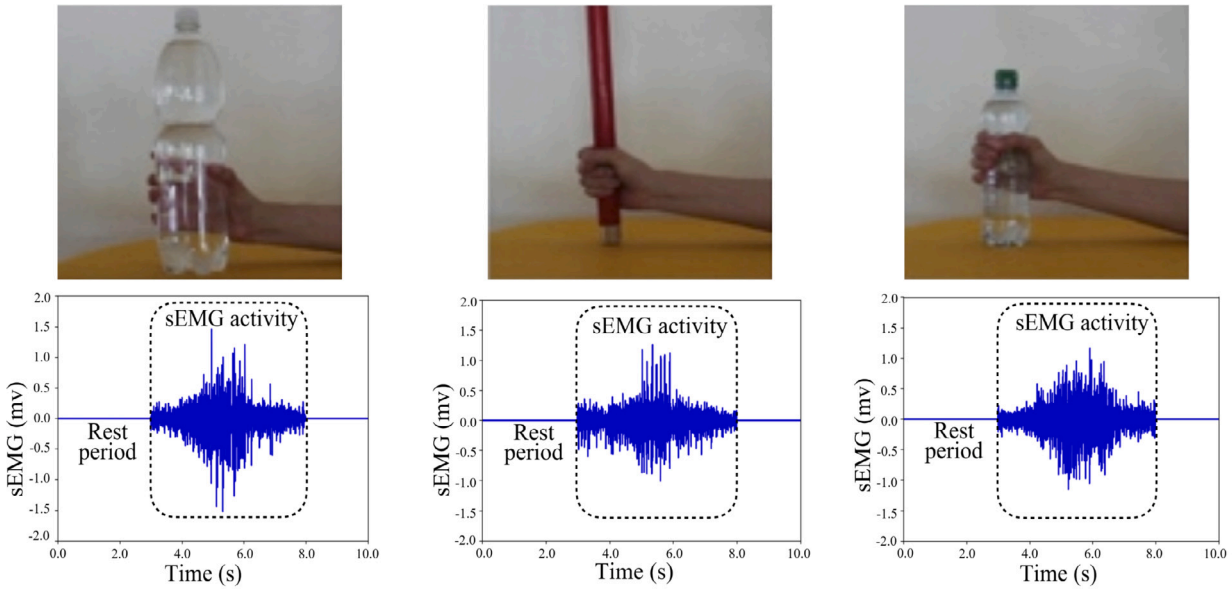


Fig. 1. Three similar gestures and their sEMG signals.

various other actions encountered in daily life. Such recognition enables the precise manipulation of virtual objects or prosthetic hands through natural hand gestures, enhancing both user experience and device functionality.

Extensive research has been conducted on the recognition of hand gestures using sEMG signals [12,23–26,28]. Hand gestures are primarily controlled by various hand-muscle contractions, and the sEMG signals are recorded by the muscle fibers during the contraction and relaxation phases [4]. Thus, current recognition methods have focused on extracting effective features and developing practical classification algorithms based on sEMG signals. However, in scenarios where there are minor differences between similar gestures (as shown in Fig. 1), which are controlled by almost identical muscle contraction patterns, recognition accuracy is normally low, regardless of whether conventional machine learning or deep learning methods are employed [35]. The main obstacle to recognizing similar gestures is the almost identical pattern of hand-muscle contractions. Thus, developing effective solutions to address this challenge is critical for enhancing the accuracy and reliability of sEMG-based gesture recognition.

Inspired by the research on muscle synergy analysis for action recognition, it is well-established that different actions are carried out through the collaborative efforts of muscles with different activation levels [16]. Even similar gestures can exhibit slight differences in muscle strength and variations in the composition of muscle synergies [3], which can be captured by sEMG electrode channels. Despite the considerable interest in this area, there has yet to be a study that analyzes channels' contribution in the recognition of similar gestures. The current feature vector combinations utilized in recognition methods do not consider this factor, and it is not easy to quantify this contribution. Additionally, current methods only explore a limited number of similar gestures and are unsuitable for real-world applications [3,35]. Moreover, deep-learning-based methods are commonly used for gesture recognition, which are typically regarded as black-box systems [11]. With these methods, we can only observe the output of a deep learning model, but it is challenging to understand why the model made a particular decision. These methods typically do not provide an in-depth understanding of recognition mechanisms and how input features contributed to the recognition results.

In this study, we devised a new Shapley-value-based solution that aims to accurately recognize and distinguish similar gestures while providing an interpretable analysis from a global perspective. Our solution addresses the challenges of current recognition methods and emphasizes the contribution of sEMG electrode channels in the recognition process. The main steps of our Shapley-value-based similar gesture recognition (SV-SGR) method are as follows. First, we develop a data-preprocessing method to transform raw sEMG signals into color sEMG images, which facilitates the use of deep learning techniques. Next, we establish a deep-neural-network (DNN)-based model for gesture recognition. Then, we devise a global explanation method based on the Shapley value to quantify each channel's contribution to recognizing similar gestures. Finally, we develop an approach to leverage the results of the interpretable analysis to provide feedback for the recognition model. To our knowledge, this is the first work on recognizing similar gestures based on Shapley values. We conducted various comparisons and interpretable analyses on real-world datasets, and the experimental results demonstrate that our method outperforms other baselines. The main contributions of this study are summarized as follows.

- This work developed a novel solution for the recognition of similar gestures, utilizing a Shapley-value-based approach. This solution highlights the impact of sEMG electrode channels in the recognition process, enabling precise identification and differentiation of similar gestures.

- This work devised a global explanation method based on Shapley values to quantify the contribution of each channel to the recognition of similar gestures. This method provides interpretable analyses to enhance our understanding of the recognition process and serves as feedback to improve recognition performance.
- This work conducted extensive comparisons on different datasets. The experimental results demonstrate the effectiveness of our method, and the accuracy improvement indicates that interpreting the results provides feedback to improve the accuracy of recognizing similar gestures.

The structure of this paper is organized as follows. Section 2 provides a review of the relevant literature. Section 3 introduces the SV-SGR method, which consists of four components: data preprocessing, DNN-based gesture recognition, global explanation, and model feedback. Section 4 presents the evaluation of our method and compare it with other baselines on different datasets. Section 5 offers valuable insights into the sEMG-based similar gesture recognition. Section 6 provides some concluding remarks and suggests future research directions.

2. Related work

This section briefly reviews the related works that closely aligns with our solution, including deep learning techniques for gesture recognition and explainable artificial intelligence approaches for action recognition.

2.1. Deep learning for gesture recognition

Deep learning techniques have gained significant attention in gesture recognition due to their ability to learn and extract high-level features automatically. Various models have been proposed for gesture recognition, including convolutional neural networks (CNNs) and recurrent neural networks (RNNs).

CNN-based methods have been widely explored for visual data processing and have shown promising results in image-based gesture recognition. Tam et al. [24] designed a gesture recognition system for multi-articulating hand prosthesis control using an embedded CNN to classify hand-muscle contractions sensed at the forearm. Molchanov et al. [18] developed a CNN-based method for recognizing hand gestures from RGB-D data, achieving a correct classification rate of 77.5% on the vision for intelligent vehicles and applications (VIVA) challenge dataset. A two-stream CNN model was also proposed by Sun et al. [23] to robustly track and recognize gestures under complex backgrounds, achieving improved detection accuracy and mean average precision. RNNs have also been widely used in gesture recognition tasks to model temporal dependencies in sequence data. Huang et al. [12] proposed a method based on bidirectional long short-term memory (Bi-LSTM) networks for recognizing sign language gestures, while Chen et al. [7] constructed a deep-bidirectional-LSTM neural network model for similar motion confusion problems in human pose recognition. Betthauser et al. [39] selected several amputee patients and healthy subjects in the NinaPro dataset. They considered sEMG decoding as a sequential modeling problem. Ergeneci et al. [40] paid special attention to movements involving the hamstrings through the attention-enhanced frequency-split convolution block. Karnam et al. [41] performed gesture recognition by extracting forward and reverse sequence information from NinaPro data as features. Rahimian et al. [42] achieved centralized localization of specific information by using attention mechanism. Model training is done by relying on the minimum training data extracted from NinaPro dataset.

Other deep learning models, such as the spatiotemporal attention mechanism proposed by Tang et al. [25] for recognizing hand gestures from video data and the deep belief network (DBN) proposed by Li et al. [15] for recognizing hand gestures from sEMG data, have also shown promising results. Ke et al. [14] intersected sEMG heatmaps and deep-learning-based gesture recognition. Godoy et al. [43] devised temporal multi-channel transformers and vision transformers, comparing their outcomes in terms of accuracy and speed of motion decoding. This advancement has shown significant improvements in the performance of muscle-machine interfaces. Dere et al. [44] evaluated the ViT model with and without an attention mechanism for the precise decoding of motor intent. Their investigation involved the examination of various input features and the integration of convolutive blind source separation (BSS) preprocessing. Hand gesture recognition in myoelectric-based prosthetic devices is a key challenge to offering practical solutions to hand and lower-arm amputees. Luo et al. [17] developed a new solution that combined synergistic myoelectrical activities of forearm muscles to improve the accuracy of recognizing gestures.

However, there is still limited research on developing effective models for similar gesture recognition in myoelectric-based prosthetic devices, which remains a critical challenge for offering practical solutions to hand and lower-arm amputees. We conducted preliminary experiments to illustrate the limited accuracy of similar gesture recognition and to emphasize the model's vulnerability to misclassification among similar gestures. The preliminary experiments in our study have demonstrated that the average recognition accuracy for three similar grasping and functional movements is approximately 58.33% for a CNN model. The range of the accuracies for the three gestures is approximately 10.05%, 8.37%, and 31.21%, respectively. These experimental results demonstrate the low accuracy and significant variation in recognition results for similar gesture recognition when using the CNN model. Moreover, the confusion matrix (shown in Fig. 2) corresponding to partial data reveals a notable misidentification rate among these similar gestures. Specifically, Fig. 2 illustrates the confusion matrix for the three gestures with the highest probability of being recognized out of the 23 gestures. Within instances of misclassification by the model, these errors typically occur between gestures that have significant similarity.

Despite the promising results achieved by deep learning methods in gesture recognition tasks, their lack of interpretability has been a subject of criticism [10]. The inability to understand the reasoning behind the model's predictions hinders its practical

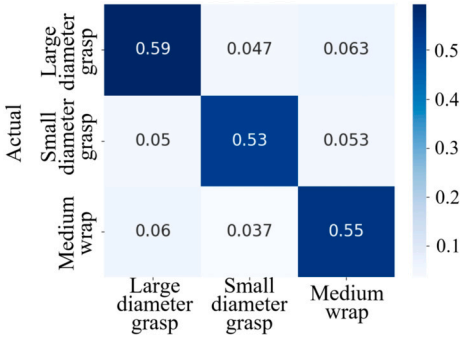


Fig. 2. Confusion matrix for three similar gestures.

applications in domains where interpretability is crucial. To address this issue, researchers have proposed various methods for interpreting deep learning models. These methods provide insights into how a deep learning model makes predictions and enhances its transparency and interpretability.

2.2. Explainable artificial intelligence for action recognition

The application of artificial intelligence (AI) systems has become increasingly widespread, using explainable AI (XAI) technologies to explain why machine learning models make specific predictions as important as the accuracy of the predictions, because it ensures the trust and transparency in the models' decision-making process [30]. XAI aims to develop models and methods that provide transparent and interpretable explanations of their decisions on a local or global level [1]. Local interpretability focuses on explaining the impact of a single instance on the model's predictive behavior, while global interpretability provides a more comprehensive understanding of the model's overall structure.

XAI has shown significant potential in improving the interpretability and transparency of action recognition models. Chakraborty et al. [5] proposed a heterogeneous recurrent spiking neural network (HRSNN) with unsupervised learning for spatiotemporal classification of video activity recognition tasks on RGB and event-based datasets. Zhang et al. [33] established a spatiotemporal attention model to identify the most discriminative regions and frames of a video for action recognition. Their visualization technique generated heatmaps to explain the attention mechanism, and the attention model significantly improved recognition accuracy while providing interpretable explanations. Another approach to XAI for action recognition is by using saliency maps. Zhang et al. [34] proposed a deep learning model integrating saliency maps to recognize actions in first-person videos. Their saliency map highlights the most important regions and frames for action recognition, significantly improving accuracy and providing explanations.

Recent works have also explored graph-based methods [8], adversarial attacks and defense mechanisms [21] to improve the interpretability of action recognition models. However, these works require further research to design an effective XAI method for practical applications. Moreover, these studies ignore the impact of channel contributions in recognizing similar gestures, and it triggers an ongoing discussion in the literature for gesture recognition [22,29]. Table 1 summarizes the related work around this area, including models, datasets, and interpretability analysis. Shapley values are a game-theoretic tool that allocates credit among players in coalitional games. It is regarded as one of the most effective solutions for cooperative game problems, as it guarantees a fair and reasonable distribution of rewards [6]. Thus, our study aims to integrate channel contributions with the method of recognizing similar gestures to develop a novel gesture recognition solution that addresses the challenges above.

3. Shapley-value-based similar gesture recognition method

This section presents the details of our SV-SGR method, which differs from conventional gesture recognition approaches by incorporating deep learning and global explanation analysis based on Shapley values. By leveraging these techniques, SV-SGR achieves high performance in recognizing similar gestures and provides a comprehensive understanding of the global importance of sEMG electrode channels. Fig. 3 provides an overview of the SV-SGR method, and the specific methodology is detailed in the following sections. First, Section 3.1 details the data-preprocessing technique for sEMG signals. Then, Section 3.2 presents a similar gesture recognition model based on DNN that employs processed sEMG color images. Section 3.3 discusses the global explanation approach based on Shapley values for gesture recognition. Finally, Section 3.4 describes how the explanation analysis is utilized to provide feedback on recognition results.

3.1. Data preprocessing

sEMG is one type of biological signal that exhibits non-stationarity, low signal-to-noise ratio, and multiple channels [19]. Each channel of sEMG signals reflects a synthesis of muscle activities in its surrounding area, and effectively extracting features in these signals is key for gesture recognition. To address these challenges, we convert sEMG signals into three-dimensional sEMG color images, which take into account these signal characteristics [9]. This conversion transforms the signal from a simple representation

Table 1
Summary of deep learning and XAI-related methods for gesture recognition.

Reference	Classification Model	Dataset	Interpretability Analysis
Chen et al. [7]	Deep-Bidirectional-LSTM	6-class kinematics data and human skeletal data fusion	None
Huang et al. [12]	CNN-Bi-LSTM	SLR500 dataset	None
Ke et al. [14]	Visual geometry group (VGG)	12-class sEMG heatmaps	None
Li et al. [15]	Deep belief network (DBN)	NinaPro dataset, exercises A, B, and C	None
Luo et al. [17]	Non-negative matrix factorization-SVM	5-class sEMG signals	None
Molchanov et al. [18]	3D-CNN	VIVA dataset, 19-class sEMG images	None
Panchoil et al. [20]	Linear discriminant analysis (LDA)	NinaPro dataset	None
Sun et al. [23]	Two-Stream CNN	36-class semantic gesture images	None
Tam et al. [24]	Embedded CNN	8-class, high-density sEMG	None
Tang et al. [25]	Selective spatio temporal feature learning	SKIG dataset, IsoGD dataset, and EgoGesture dataset	None
Zabihi et al. [31]	Transformer for hand gesture (TraHGR)	NinaPro dataset 2, exercises B and C	None
Zheng et al. [37]	Long short-term memory (LSTM) network	NinaPro dataset 1, exercise C	None
Zhou et al. [38]	Random forest (RF)	NinaPro dataset 4	None
Betthauser et al. [39]	Temporal convolutional network (TCN)	NinaPro datasets 2 and 3, Capgmyo dataset, exercise C	None
Ergeneci et al. [40]	CBAM-CNN	NinaPro dataset 2, 49-class sEMG signals	None
Karnam et al. [41]	CNN-Bi-LSTM	NinaPro dataset, BioPatRec dataset, and UCI gesture dataset	None
Rahimian et al. [42]	Attention-TCN	NinaPro datasets 2 and 5, 53-class sEMG signals	None
Chakraborty et al. [5]	Heterogeneous recurrent spiking neural network (HRSNN)	UCF101 dataset	Heterogeneous spike-time-dependent-plasticity (STDP)
Ding et al. [8]	Temporal segment graph convolutional network (TS-GCN)	NTU-RGB+D dataset with 3D human skeletal data	Visualization of adaptively constructed temporal graphs
Ren et al. [21]	Generative adversarial network (GAN)	Images of various types of counterattacks	Adversarial attacks and defense mechanisms
Siemers et al. [22]	SVM and RF	Data on various biological enzymes	Importance analysis of learning characteristics
Watson et al. [29]	Extreme gradient boosting (XGBoost)	MNIST dataset	Games with modified eigenfunctions

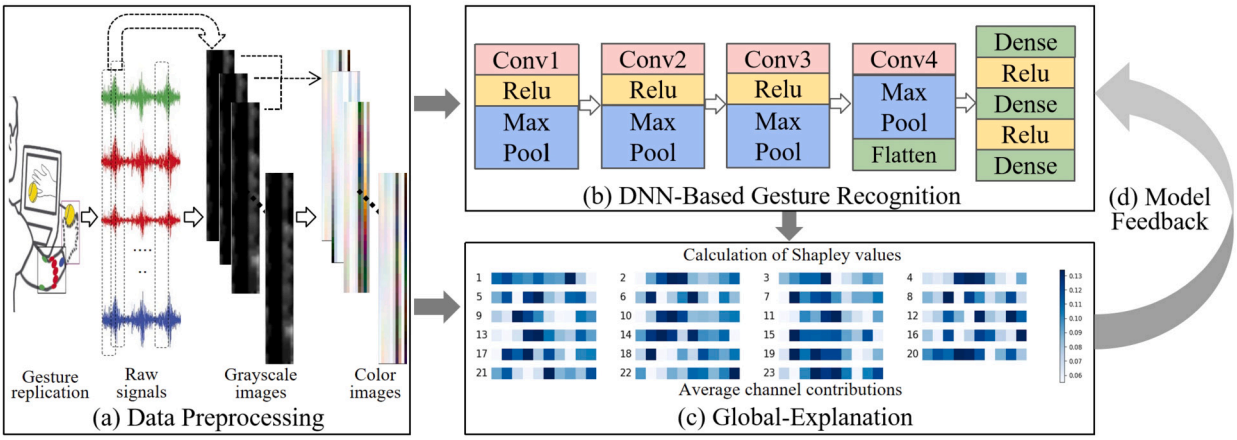


Fig. 3. The solution framework.

of temporal information into an image. The patterns inherent in sEMG signals can be notably intricate, particularly during the identification of similar gestures. Representing the data in image format enables the model to grasp and identify these intricate patterns at a more sophisticated level and from a broader perspective, especially when dealing with signals involving multiple muscle groups and complex muscle synergies.

In the SV-SGR method, the data-preprocessing step [Fig. 3(a)] involves converting sEMG signals into sEMG color images. Specifically, we attach N sEMG sensors to the forearm to collect N channels of sEMG signals, resulting in a $TF \times N$ matrix, where T is the sampling duration, and F is the sampling frequency. The column number of this matrix represents the electrode number corresponding to an sEMG sensor, and the number of rows is the product of the sampling duration and frequency, TF. The signal collected by each electrode undergoes filtering and is then assigned a label corresponding to the gesture performed. Each row is labeled i, representing the gesture i performed at that moment.

To detect activity segments in sEMG data, we divide them into action modes ($i \neq 0$) and rest modes ($i = 0$) based on the label i. We scan the label of each row and classify the data labeled $i = 0$ as a rest mode, while the data corresponding to non-zero labels is classified as an action mode.

We set an $M \times N$ analysis window to analyze the sEMG data. Such windows can be either overlapping or non-overlapping. In practical applications, overlapping analysis windows are commonly used to increase the number of available samples. We select only the action modes that are labeled as non-zero as temporal arrangements, and use an overlapping window with a sliding step size of s. An $M \times N$ array with the same gesture label i is considered valid data, with the label of this array defined as i. The number of arrays, K, can be calculated as

$$K = \left\lfloor \frac{TF - M}{s} \right\rfloor + 1 \tag{1}$$

Where $\lfloor k \rfloor$ represents the largest integer less than or equal to k.

Based on the steps above, each array is converted into a $M \times N$ matrix A_k, where $k \in (1, K)$. We then normalize the matrix A_k and map its values to the range of (0, 255), which enables convenient and efficient data processing.

$$A_k^* = \left(\frac{A_k - \min A_k}{\max A_k - \min A_k} \right) \times 255 \tag{2}$$

The valid array data with gesture label i are transformed into K grayscale images, each with a size of $M \times N$. To improve the accuracy of similar gesture recognition, we combine three adjacent grayscale images with the same gesture label to form a three-dimensional color image. This step results in a dataset of sEMG color images, which are used in the subsequent steps of the SV-SGR method.

3.2. DNN-based gesture recognition

To recognize similar gestures using sEMG data, we have developed a DNN model [Fig. 3(b)] comprising convolutional layers, fully connected layers, and a softmax layer. This model takes sEMG color images as input and leverages deep learning techniques to recognize patterns within these images. The recognition process involves two stages: an offline training stage and an online recognition stage. During the training stage, we use sEMG color images and their corresponding gesture labels to train a classifier that can accurately predict the gesture associated with each image. The trained classifier is then used to recognize gestures through the sEMG images in the recognition stage. The primary parameters and corresponding descriptions are summarized in Table 2.

In this study, we adopt an image-wise deep-CNN architecture to extract features from sEMG color images. Our CNN architecture comprises four convolutional layers with a uniform kernel size of 3×3, which produce 32, 64, 128, and 128 feature maps, respectively. This architecture effectively captures features within specific regions of the sEMG image using a set of translation-invariant filters.

The l-th convolutional layer takes the activation $a^{(l)}$ and produces the output $O^{(l)}$ by convolving $a^{(l)}$ with the convolutional kernel $w^{(l)}$. The receptive field size of the convolutional kernel is $A \times B$, and the sliding length is z. The bias term is denoted as $\beta^{(l)}$. The convolutional layer performs the calculation as

$$O_{xy}^{(l)} = \sum_{a=1}^{A} \sum_{b=1}^{B} \left(\omega_{ab}^{(l)} * \alpha_{(zx+a)(zy+b)}^{(l)} + \beta^{(l)} \right) \tag{3}$$

where x represents the x-th row in the image, y represents the y-th column in the image, and the operator * denotes the convolution operation.

After the convolutional layers, we employ a flatten layer with 768 neurons, a dense layer with 1024 neurons, and another dense layer with 512 neurons to facilitate the conversion of feature maps into a new feature representation. These fully connected layers serve as input to a softmax layer, which transforms the hidden state in the dense layer into prediction probabilities for gesture recognition. The cross-entropy-based loss function used in the model is defined as

$$L = -\frac{1}{batch\_size} \sum_{r=1}^{batch\_size} \sum_{i=1} p_{ri} \log q_{ri} \tag{4}$$

where p_{ri} is the actual output probability for the gesture i at batchsize r, and q_{ri} is the expected output probability for the gesture i at batchsize r. This loss function is used to train the model during the offline training stage, ensuring that the classifier can accurately predict each gesture associated with an sEMG color image.

Table 2
The description of model parameters.

Parameter	Description
l	Network layer l
$O^{(l)}$	Output of layer l
A	Length of a convolution kernel
B	Width of a convolution kernel
$A \times B$	Size of the receptive field for a convolutional kernel
$\omega^{(l)}$	Convolutional kernel for layer l
z	Sliding length
x	The x-th row in an image
y	The y-th column in an image
$\beta^{(l)}$	Bias term for layer l
i	Gesture type
p_{ri}	Actual output probability for the gesture i at batchsize r
q_{ri}	Expected output probability for the gesture i at batchsize r

3.3. Global explanation

In this subsection, we introduce an explainable method based on the Shapley value for similar gesture recognition. This method provides an effective means for understanding the contribution and significance of input features in recognizing similar gestures. The recognition results can be explained through game theory by considering each sEMG electrode channel as a "player" in the game. The Shapley value, which represents the weighted average of marginal contributions for a channel across all possible coalitions, can be used to elucidate these results [6].

First, we calculate the total contribution of the sEMG data acquired from all input electrode channels in the task of similar gesture recognition. Specifically, the total contribution for the task is evaluated as the discrepancy between the output $f(C)$ obtained by inputting all channels' signals into the recognition model and the output $f(\emptyset)$ obtained when there is no input.

$$\left| f(C) - f(\emptyset) \right| = \sum_{n=1}^{N} \varphi_n \tag{5}$$

where $\emptyset$ denotes zero input, and φ_n denotes the Shapley value of the n-th input channel.

Then, the total contribution obtained by N input electrode channels must be "fairly" allocated to each channel. In this study, we adopt the Shapley value as a metric to measure the impact of each channel on gesture recognition performance when the channel is either participating or not participating in the game. The Shapley value of each channel is computed by weighting and aggregating all probable combinations of electrode channels. Based on this, the Shapley value of each channel denotes its contribution to the task of similar gesture recognition and can be calculated as

$$\varphi_n = \sum_{S \subset N \setminus \{n\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} \left(v(S \cup \{n\}) - v(S) \right) \tag{6}$$

where N is the set of all electrode channels, S is the set of channels involved in game interactions, $v(S)$ is the contribution value of S, and $v(S \cup \{n\}) - v(S)$ denotes the marginal contribution of the channel n to participating in S.

In the task of recognizing similar gestures, the process of calculating Shapley values is determined by the game interaction between sEMG electrode channels. To ensure an accurate estimation of the contribution of each channel, we employ a permutation and combination technique that guarantees the coverage of all interactions between channels during the game. Specifically, we carry out computations for all possible combinations of lengths ranging from 1 to N, where N is the total number of electrodes. Each combination corresponds to one round of game interaction.

For each round of game interaction, we set the input electrode channels based on the corresponding combination. The channels that are part of the combination remain unchanged, while the channels that are not included are set to 0. We then use this input data for gesture recognition and retrieve the prediction matrix for the recognition task before applying the softmax layer in the DNN. This matrix comprises the scores for different channels in this round, which serve as the contribution value $v(S)$.

After computing all feasible combinations and contribution values, we calculate the Shapley value of each channel for gesture recognition based on Eq. (6). The Shapley value of each channel reflects its contribution to the task of recognizing similar gestures, taking into account all possible coalitions of channels and their marginal contributions. By computing the Shapley value for each channel, we can obtain a better understanding of the relative importance of each channel in the recognition task.

Finally, we obtain a Shapley value array for each type of gesture, and all arrays of gestures constitute a matrix. By analyzing the Shapley values in this matrix, we can achieve a global explanation for the relationship between input channels and similar gesture recognition. The contribution of each channel involved in recognizing gestures can be calculated as

$$I_n = \frac{1}{X} \sum_{j=1}^{X} \left| \varphi_n^{(j)} \right| \tag{7}$$

where I_n denotes the average Shapley value for the n-th input channel, which indicates its contribution to gesture recognition. X denotes the number of samples in the dataset. $\varphi_n^{(j)}$ represents the Shapley value for the n-th input channel, which is determined by analyzing the j-th sample. Through the above steps, we ultimately analyze the contribution of each input channel in similar gesture recognition. The accuracy of the calculation results can be verified via Eq. (5).

3.4. Model feedback

This subsection describes how we leverage explanation analysis to provide feedback for the DNN-based recognition model. We use the contribution ratio based on Shapley values to enhance the performance of recognizing similar gestures. After the global explanation analysis, the contribution of the input channel n can be computed and subsequently normalized [as Eq. (8)] to obtain the contribution ratio C_n for each gesture. This ratio reflects the varying importance levels of different electrode channels in the similar gesture recognition task.

$$C_n = \frac{\varphi_n}{\sum_{t=1}^{N} \varphi_t} \tag{8}$$

In this study, we enhance the recognition model by incorporating the contribution ratio of each electrode channel as a weight for data augmentation. To achieve this, we apply this method to the first convolutional layer of the deep neural network, as shown in Eq. (9). By incorporating weights into the convolutional kernel, we can enhance the data obtained from important electrode channels, which can guide the recognition model to focus on learning this segment of the data. This adjustment effectively shifts the focus of the recognition model during the convolution process. Notably, we leverage the contribution ratio as feedback to the input section of the model, thereby improving the overall performance of the recognition model.

$$O_{xy}^{(l)} = \sum_{a=1}^{A} \sum_{b=1}^{B} \sum_{n=1}^{N} \left(C_n + \omega_{ab}^{(l)} \right) * \alpha_{(zx+a)(zy+b)}^{(l)} + \beta^{(l)} \tag{9}$$

Moreover, model feedback requires the fine-tuning of model parameters. In the fine-tuning stage, all convolutional layers are fixed, and only the last fully connected layers are adjustable. This ensures that only a small fraction of the parameters are optimized during the retraining process. The resulting recognition model, obtained through this training process, effectively improves the performance of recognizing similar gestures.

4. Experiments

To evaluate the performance of the SV-SGR method under various experimental conditions, we conducted a series of experiments on three publicly available datasets, namely Non-Invasive Adaptive Prosthetics (NinaPro) DB1, DB2, and DB3 [2]. This section first introduces the basic information of these datasets, then provides the experimental settings and comparison methods, and finally demonstrates the experimental results and performance analysis.

4.1. Dataset description

The NinaPro dataset is a widely used benchmark dataset in the field of sEMG-based gesture recognition, aimed at advancing the research on hand myoelectric prosthetics. The sEMG signals undergo filtering prior to segmentation, a critical step designed to enhance the quality, efficiency, and precision of subsequent signal processing, while mitigating the impact of noise and edge effects. Moreover, to ensure the fairness and comparability of the experiments, we have used the same data processing methods as in the related experiments. In this study, we utilized the three sub-datasets (DB1, DB2, and DB3), comprising a total of 78 participants, out of which 67 were intact subjects and 11 were trans-radial amputated subjects. For the exercise C section of the NinaPro dataset, 23 grasping and functional movements have been included. Similar gestures are delineated based on visual resemblances encompassing the gesture's shape, finger arrangement (with particular emphasis on the positioning of the thumb relative to the other fingers), and the overall contour outline of the gesture. Moreover, similar muscle activation patterns and intensities are identified in these gestures through the analysis of sEMG signal maps. Feix et al. [46] introduced a classification criterion for grasping gestures which entails defining fingers with similar orientations and angles. Notably, all 23 grasping gestures exhibit a virtual finger count of 2, with each gesture sharing identical gesture shapes and finger arrangements with specific other gestures. The datasets were carefully designed to include similar gestures performed by participants, enabling us to evaluate the performance of our SV-SGR method under diverse experimental conditions. Table 3 summarizes essential information for the three datasets.

The DB1 dataset employs 10 Otto Bock sEMG electrode channels with a sampling frequency of 100 Hz. Among these channels, eight (channels 1-8) are evenly distributed around the forearm at the level of the radio-humeral joint, while the remaining two electrodes (channels 9 and 10) are attached to the primary activity spots of the flexor and extensor digitorum superficialis. The DB2 and DB3 datasets utilize 12 Delsys sEMG electrode channels with a higher sampling frequency of 2000 Hz. Ten of these channels are identical in placement to those in the DB1 dataset, while the remaining two electrodes (channels 11 and 12) are attached to the primary active points of the biceps and triceps brachii. It is worth noting that the DB1 and DB2 datasets comprise data collected from healthy subjects, whereas the DB3 dataset comprises data collected from amputee subjects.

Table 3
Statistics of three NinaPro sub-datasets.

	DB1	DB2	DB3
Intact subjects	27	40	0
Trans-radial amputated subjects	0	0	11
Sampling frequency (Hz)	100	2000	2000
Number of similar gestures	23	23	23
sEMG electrodes	10 Otto Bock	12 Delsys	12 Delsys
Number of input channels	10	12	12

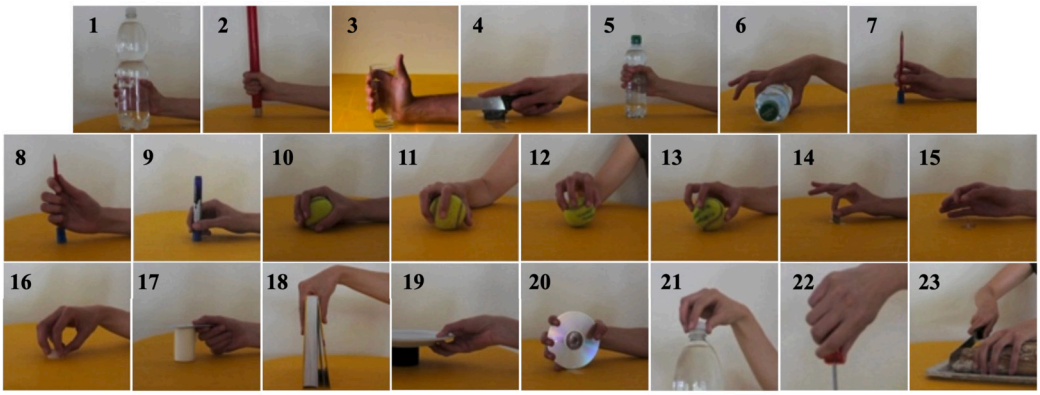


Fig. 4. Grasping and functional movements.

The primary focus of this study is to recognize similar gestures, specifically Exercise C in the NinaPro datasets. Exercise C comprises 23 grasping and functional movements (as depicted in Fig. 4) that cover most similar hand movements in daily activities. These movements are numbered from 1 to 23, and more detailed information can be found in [2]. During the experiment, participants were instructed to replicate the movements displayed on a computer screen, while repeating each action for 5 seconds, with intermittent rest periods of 3 seconds. These datasets not only facilitate the comparison of gesture recognition performance under diverse experimental conditions, including different sampling rates and subjects with or without limb amputations, but also enable us to conduct an interpretable analysis of gesture recognition across these conditions.

4.2. Experimental settings and comparison methods

We implemented all algorithms in Python and conducted computational experiments on a machine with 128 GB RAM, Intel(R) Xeon(R) Gold 6128 CPU @ 3.40 GHz with an NVIDIA GeForce RTX 2080 Ti GPU. The deep learning methods were implemented based on the TensorFlow[1] framework. In this study, the recognition accuracy is defined as the ratio between the number of correctly recognized segments (N_c) and the total number of testing segments (N_t). This metric is preferred over global accuracy as it provides a more detailed evaluation of the performance of the recognition method for each individual gesture [45]. The accuracy of recognizing a target gesture i, Acc_i, is calculated as

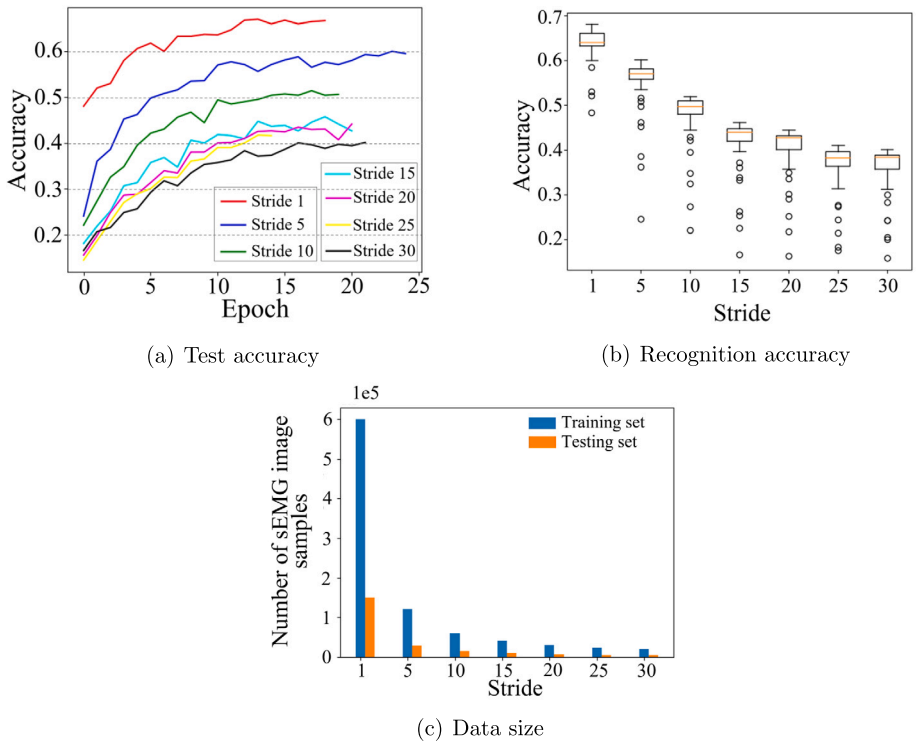
$$Acc_i = \frac{1}{I}\sum_{i=1}^{I}\left[\frac{N_c}{N_t}\right]_i \tag{10}$$

where I represents the total number of gesture types. A higher accuracy indicates that this method performs better in recognizing gestures.

The size of the analysis window and the sliding step size, also known as stride, are crucial parameters that can significantly impact the accuracy of gesture recognition. While a larger analysis window size is generally considered to improve recognition accuracy, it may also lead to poor efficiency. Moreover, it is essential to maintain a response time of 300 ms in practical applications, and any significant delay can significantly impair system effectiveness. Thus, the analysis window size is typically set within the range of 200-300 ms [13]. Similarly, the sliding step size must be chosen reasonably. If the stride is set too high, non-overlapping analysis windows will be generated, resulting in insufficient data for accurate recognition. Conversely, setting the stride too small will produce a large amount of redundant data, which leads to poor efficiency.

To investigate the impact of different strides on model performance, we conducted comparisons using OttoBock's 10-channel sEMG data (DB1) collected at a sampling frequency of 100 Hz and Delsys's 12-channel sEMG data (DB2 and DB3) acquired at

[1] https://tensorflow.google.cn/.



(a) Test accuracy (b) Recognition accuracy

(c) Data size

Fig. 5. Impact of different strides on model performance.

a sampling frequency of 2000 Hz. The NinaPro dataset has undergone several signal processing steps, including normalization, synchronization, relabeling, and filtering. Synchronization involves using linear interpolation (for real-valued streams) or nearest-neighbor interpolation (for discrete streams) to upsample all streams to the highest sampling frequency (either 2 kHz or 100 Hz, depending on the type of sEMG electrodes used). The electrodes are not affected by power line interference, which can impact the encoded signals in certain circumstances. Thus, before synchronization, the sEMG signals were processed to remove 50 Hz power-line interference using a Hampel filter. We analyzed changes in test accuracy by using early stopping [Fig. 5(a)], the accuracy of recognizing similar gestures [Fig. 5(b)], and the amount of data in the training and testing sets [Fig. 5(c)] with different strides. As we varied the stride from 1 to 5, we observed a reduction in the number of sEMG image samples. Specifically, the number of samples decreased to one-fifth of the sample size when the stride was set to be 1. This reduction in sample size resulted in a drop in accuracy of approximately 6%. To trade off the recognition efficiency and accuracy, we set the analysis window of DB1 to be 30×10 and the stride to be 5. For the DB2 and DB3 datasets, we employed a similar analysis method and set the analysis window size to be 400×12 and the stride to be 30. These settings ensured that the recognition model had access to sufficient data to accurately classify different gestures while maintaining a reasonable computational efficiency.

Before determining the DNN model we would employ, we conducted an ablation study based on classical recognition models [15]. We set the baseline of DB1C to a four-layer convolutional architecture. The DB2 and DB3 datasets have a much higher sampling rate (20 times higher) than DB1, resulting in a significant increase in the amount of data. The original four-layer CNN model's complexity is insufficient to capture patterns within the datasets, potentially hindering the acquisition of crucial features and resulting in model underfitting, consequently restricting performance enhancement. Thus, the original four-layer convolutional neural network model struggled to effectively train on these datasets. To improve the performance of gesture recognition, we added two additional convolutional layers after the fourth layer of the original model for the DB2 and DB3 datasets. Adding convolutional layers facilitates the model's ability to learn more complex feature representations. Consequently, the Baseline for both DB2C and DB3C was established with six layers. Following each convolutional operation, the architecture incorporates ReLU activation functions and max-pooling layers. This is followed by three fully connected layers and a softmax layer. For Ablation 1, one convolutional layer is removed to assess the impact on model performance. In Ablation 2, two additional convolutional layers are eliminated. Lastly, Ablation 3 involved replacing the activation function from ReLU to Sigmoid to evaluate the effect of this change on the model's predictive capabilities. As shown in Table 4, the Baseline exhibit the best performance based on metrics such as Accuracy, Recall, and F1 Score. We used 80% of the sEMG color image data for training, while the remaining 20% was used for testing. Our primary focus is on inter-subject evaluation, where the experimental data comprise sEMG images derived from different individuals. During the evaluation phase, these images are matched against fresh sEMG data retrieved from an entirely different individual. Thus, the experiment is inter-subject evaluation.

In this study, the feedback optimization on DNN was performed using the Adam optimizer, and the loss function was chosen as sparse categorical cross-entropy. The Adam optimizer, which combines the advantages of momentum and the RMSProp algorithm, is

Table 4
Results of ablation experiments.

	Accuracy	Recall	F1 Score
DB1C Baseline	59.39%	0.60	0.59
DB1C Ablation 1	53.33%	0.52	0.53
DB1C Ablation 2	37.13%	0.37	0.37
DB1C Ablation 3	55.22%	0.56	0.55
DB2C Baseline	67.87%	0.71	0.68
DB2C Ablation 1	62.26%	0.62	0.62
DB2C Ablation 2	47.35%	0.48	0.42
DB2C Ablation 3	64.32%	0.64	0.64

an adaptive learning rate optimization algorithm. The process of network training illustrates the advantages of Adam's optimizer [47]. It dynamically adjusts the learning rate of each parameter by computing the first moment estimate (i.e., the mean of gradients) and the second moment estimate (i.e., the mean of squared gradients), thereby minimizing the loss function [Eq. (4)]. The model feedback in SV-SGR involves several steps. First, a DNN-based model for gesture recognition is constructed. Then, based on the recognition results, an interpretable analysis is conducted using a game-theoretic interaction approach. The contribution ratio (C_n) is derived from the interpretable analysis, and C_n is incorporated into the convolution kernel to form a feedback, as shown in Eq. (9). Finally, the convolutional kernel parameters of the model are adjusted using the obtained interpretability results, followed by retraining of the model to enhance its recognition performance. The batch size was set to be 32. In additional, we implemented early stopping to prevent overfitting and ensure that the model was trained only until a satisfactory level of performance was achieved.

In this study, we compared the SV-SGR method with six representative methods, including conventional machine learning methods (LDA, SVM, and RF) and deep learning methods (CNN, LSTM, and TraHGR). Further details regarding these comparison methods are elaborated below.

- Linear discriminant analysis (LDA): Set marginal discrete wavelet transform (mDWT) as the selected signal feature and LDA as classifier [20].
- Support vector machine (SVM): Set root mean square (RMS) as the selected signal feature and SVM as classifier [22].
- Random forests (RF): Set a normalized combination of mDWT, RMS, and Histogram as the selected signal feature and RF as classifier [38].
- Convolution neural network (CNN): Set a deep neural network architecture to extract features and CNN as classifier [15]. The network setting is the same as the SV-SGR method before model feedback.
- Long short-term memory network (LSTM): Set a recurrent neural network architecture to learn long-term spatiotemporal features. The network setting is the same as that used in a previous study [37].
- Transformer for hand gesture recognition (TraHGR): Set a hybrid framework based on the transformer architecture as classifier, which is a new deep learning model. The network setting is the same as that used in another study [31].

By comparing the performance of these methods, we aim to evaluate the effectiveness of the SV-SGR method and explore its potential advantages and limitations.

4.3. Experimental results and performance analysis

To validate the effectiveness of the SV-SGR method, we first conducted a comparative analysis of its recognition results against those of several conventional machine learning methods and deep learning methods across various datasets. We then performed a specific interpretable analysis for the results of recognizing similar gestures. Finally, we compared the recognition accuracy for 23 similar gestures obtained from the original model (DNN) and the feedback-enhanced model (SV-SGR). We conducted an analysis focused on the accuracy improvement for the DB1 and DB2 datasets at different sampling frequencies, as well as for the DB2 and DB3 datasets with different experimental subjects. By conducting these analyses, we aimed to comprehensively evaluate the performance of the SV-SGR method and to demonstrate its potential to achieve high accuracy in recognizing similar gestures across various datasets and experimental conditions. The SV-SGR method enhances the accuracy of recognizing similar gestures comparing to conventional machine learning algorithms. Moreover, it ensures that the outcomes of recognition are interpretable. Although it does not possess a clear advantage over complex deep learning models in terms of recognition accuracy, SV-SGR stands out for its strong interpretability, coupled with a more simple architectural framework.

4.3.1. Recognition accuracy for similar gestures

In this study, we conducted a comparative analysis on the average accuracy of recognizing similar gestures across different datasets (as shown in Table 5). We implemented the proposed preprocessing pipeline to the comparison architectures, as detailed in Subsection 4.2. Our experimental results demonstrate that the SV-SGR and TraHGR methods perform best in all three datasets. However, TraHGR consists of a TNet path and an FNet path in parallel, followed by a linear layer that acts as a fusion center to combine the features extracted from the two parallel paths. TraHGR also contains embedded patches, transformer encoder and many other components. The model structure is extremely complex compared to SV-SGR. Moreover, the average accuracy of SV-SGR is

Table 5
Comparisons of average accuracy of recognizing similar gestures.

	LDA	SVM	RF	CNN	LSTM	TraHGR	SV-SGR
DB1	53.79%	63.46%	68.37%	59.39%	72.85%	80.34%	79.56%
DB2	55.87%	69.28%	71.62%	67.87%	74.59%	81.44%	79.48%
DB3	35.62%	40.46%	52.16%	48.52%	52.50%	56.37%	57.69%

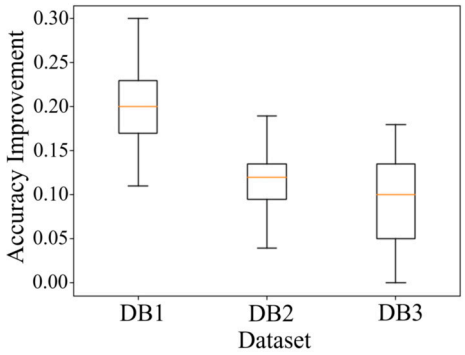


Fig. 6. Accuracy improvement between CNN and SV-SGR.

found to surpass those of conventional machine learning methods (LDA, SVM, and RF) by 15.50% and is superior to those of deep learning methods (CNN and LSTM) by 9.62%.

The accuracy of DNN-based gesture recognition varies across different datasets (Table 5). Among the three datasets, the recognition accuracy on DB1 is approximately 59.39%, while the accuracy on DB2 is only 67.87%, and that of DB3 is even lower, at 48.52%. The CNN method does not exhibit significant advantages compared to conventional machine learning methods, and its recognition accuracy is somewhat inferior to that of advanced deep learning models due to its simple CNN architecture with only four or six convolutional layers.

However, our SV-SGR method implements feedback optimization based on DNN and exhibits a clear advantage in recognition accuracy (79.56%, 79.48%, and 57.69%) over conventional machine learning methods on three datasets. Its accuracy is comparable to or slightly lower than that of advanced deep learning methods. Although the network architecture used by SV-SGR is simple, which results in a reduction in recognition accuracy compared to advanced deep learning methods, we analyzed the accuracy improvement between CNN and SV-SGR on three different datasets (Fig. 6), which demonstrated an average improvement of 20.17%, 11.61%, and 9.17%. These results highlight the significant impact of Shapley-value-based interpretable analysis in similar gesture recognition tasks (further details in the next section). Moreover, the simple network structure of SV-SGR reduces the number of training parameters and enhances the efficiency of gesture recognition. SV-SGR achieves high recognition accuracy that is not inferior to other advanced deep learning methods while ensuring efficiency, which highlights its practicality.

4.3.2. Explanation analysis for gesture recognition

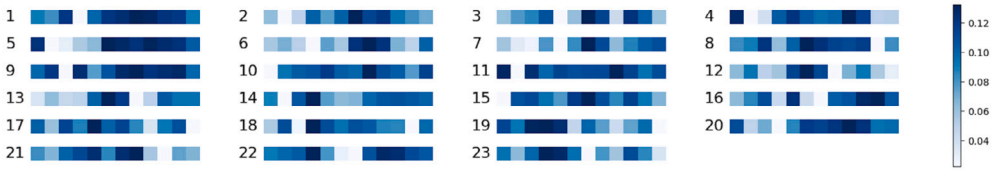
This subsection provides a global explanation of similar gesture recognition and presents visualizations to facilitate the understanding of the recognition results. Fig. 7 depicts the average channel contributions for recognizing 23 similar gestures. Each subgraph corresponds to a single gesture from the 1st to the 23rd one, and the distribution of contributions from either 10 (DB1) or 12 (DB2) channels is illustrated in each subgraph. The color map in each subgraph represents the average contribution ratio for each gesture, with darker colors indicating higher contributions. The differences in channel contributions among the gestures shown in the figure suggest that these differences can be utilized as a recognition feature, potentially leading to improved recognition results.

However, the precise computation of Shapley values depends on the recognition model's capacity to identify gestures effectively. As previously discussed, we found that the recognition accuracy in the DB3 dataset was only 48.52%, which would not accurately reflect the channel contributions if we computed the Shapley value using this recognition model. Moreover, the experimental conditions for the DB2 and DB3 datasets are the same. Thus, this study conducted an explainable analysis for all gestures in the DB1 and DB2 datasets. To this end, we used SHAP diagrams[2] to demonstrate the distinct contributions of 10 channels for three representative gestures in DB1 (Fig. 8) and 12 channels for the same gestures in DB2 (Fig. 9), which encompass the large diameter grasp (gesture 1), small diameter grasp (gesture 2), and medium wrap (gesture 5). The horizontal axis in these figures represents the average contribution ratio for recognizing a given gesture, with the color red indicating a positive contribution and playing a favorable role in the recognition process. The vertical axis represents the electrode channel number.

The explanation analysis focused on the channels with contribution ratios exceeding the average (where the average contribution ratio for 10 channels is 10.00%, and for 12 channels is 8.33%). Specifically, we considered the top three channels with the highest

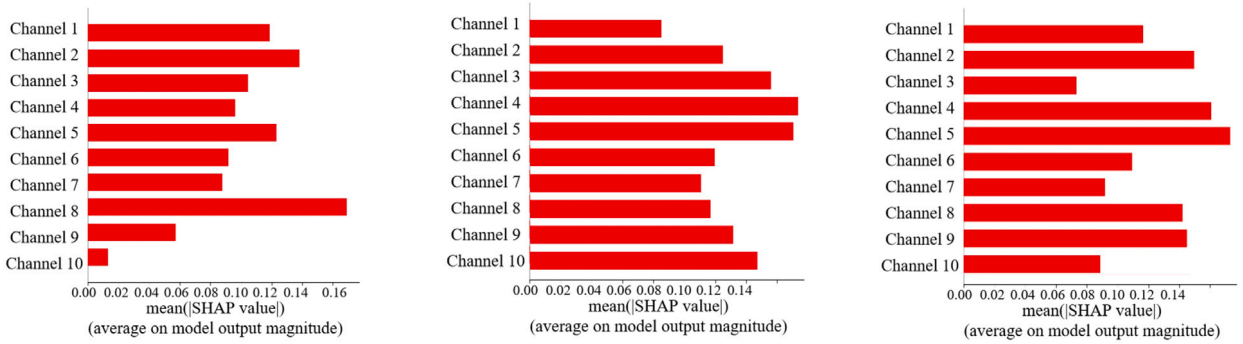
[2] https://shap.readthedocs.io/en/latest/index.html.

(a) Contributions from 10 channels in the DB1 dataset



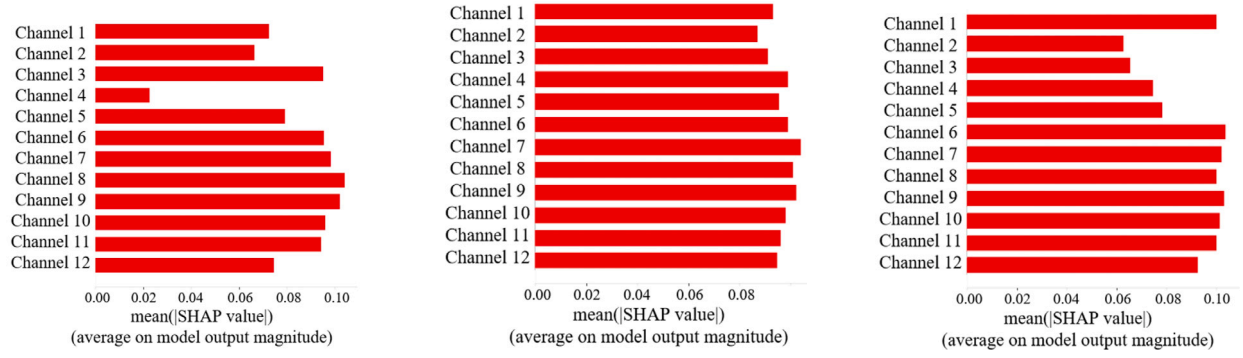
(b) Contributions from 12 channels in the DB2 dataset

Fig. 7. Average channel contributions for the recognition of 23 similar gestures, where 1, 2, ⋯, 23 are gesture labels.



(a) Large diameter grasp (b) Small diameter grasp (c) Medium wrap

Fig. 8. Channel contributions for the recognition of three similar gestures in the DB1 dataset.



(a) Large diameter grasp (b) Small diameter grasp (c) Medium wrap

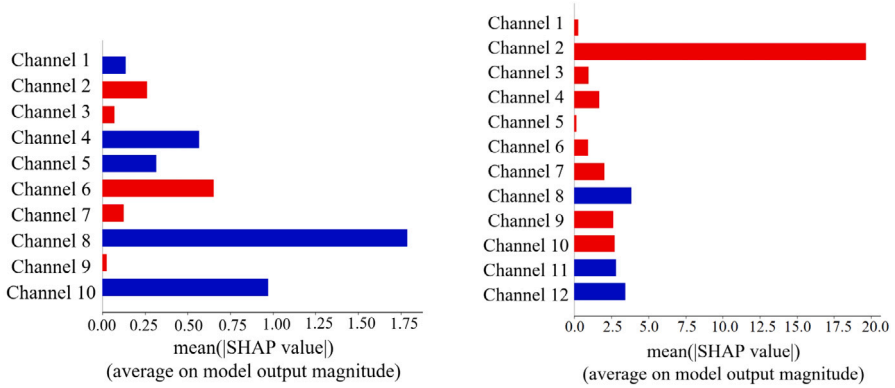
Fig. 9. Channel contributions for the recognition of three similar gestures in the DB2 dataset.

contribution and the three channels with the lowest contribution. The summarized results, as presented in Table 6, use a color scheme where gray indicates that the channel's role in recognizing the gesture is similar to its role in recognizing the other two gestures, highlighting the similarity of channel contributions. Conversely, blue indicates that the channel's role in the recognition task differs from its roles in the other two recognition tasks, emphasizing the diversity of channel contributions.

The channel contributions for recognizing similar gestures can be found to be similar but not identical. For gesture recognition based on 10 electrode channels (DB1), the recognition of gestures 1, 2, and 5 relies more on the sEMG information provided by channels 2, 4, and 5, while channels 7 and 10 contribute less to these three recognition tasks. For gesture recognition based on

Table 6
Channel contribution analysis for different gestures.

Dataset	Gesture	Contribution ratios exceed the mean	Top 3 channels	Lowest 3 channels
DB1	Gesture 1	**1**, 2, 3, 5, 8	8, 2, 5	7, **9**, 10
	Gesture 2	3, 4, 5, **10**	4, 5, 3	**8**, 7, **1**
	Gesture 5	2, 4, 5, 8, **9**	5, 4, 2	7, 10, **3**
DB2	Gesture 1	**3**, 5, 6, 7, 8, 9, 10, 11	8, 9, 7	1, 2, 4
	Gesture 2	**4**, 5, 6, 7, 8, 9, 10, 11	7, 9, 8	1, 3, 2
	Gesture 5	**1**, 6, 7, 8, 9, 10, 11, 12	6, 9, 7	4, 3, 2



(a) Sample No. 397 for gesture 5 (b) Sample No. 14 for gesture 1

Fig. 10. Instances of different channel contributions for recognizing similar gestures.

12 electrode channels (DB2), the recognition of gestures 1, 2, and 5 relies more on the sEMG information provided by channels 7, 8, 9, and 11, while channels 1, 2, 3, and 4 have a lesser contribution to the three recognition tasks. These variations in channel contributions can be utilized as feature information by the recognition model to distinguish similar gestures.

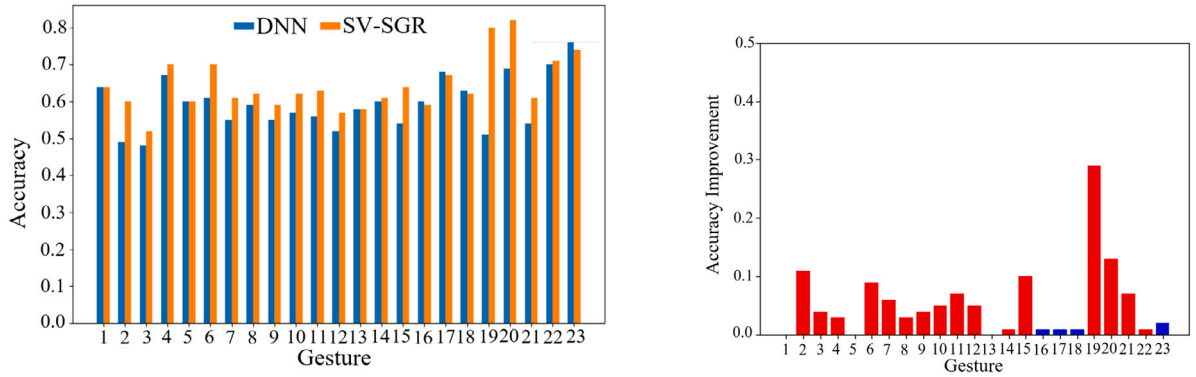
During the explanation analysis, it is also possible to observe negative values in the SHAP diagram, as illustrated in Fig. 10. The red color in the figure indicates the channel's positive contribution to gesture recognition, while blue shows its negative contribution. This finding suggests that recognition models, such as deep learning, may be affected negatively by the information provided by these channels, thereby reducing recognition accuracy. It also suggests that the sEMG information provided by electrode channels does not always have a positive effect and may be redundant or even negatively affect the model's judgment.

4.3.3. Accuracy improvement in DB1 dataset

After conducting the explanation analysis in the DB1 dataset, we utilized the model feedback method (provided in Section 3.4) to enhance the accuracy of the recognition model. We compared the accuracy of recognizing 23 similar gestures before and after feedback enhancement using the Shapley value obtained with a 30×10 analysis window, as shown in Fig. 11(a). The blue bar in the figure represents the recognition accuracy of the original model (DNN), while the orange bar represents the recognition accuracy of the feedback-enhanced model (SV-SGR). The results demonstrate that SV-SGR achieves a significant improvement in recognition accuracy compared to the original model. Moreover, Fig. 11(b) shows the specific improvement of each gesture after feedback optimization. The blank areas in the figure indicate almost no effect, red areas indicate improvement, and blue areas indicate negative impacts (decreasing recognition accuracy). These results demonstrate that SV-SGR effectively improves recognition accuracy and highlights the importance of obtaining accurate channel contributions through explanation analysis.

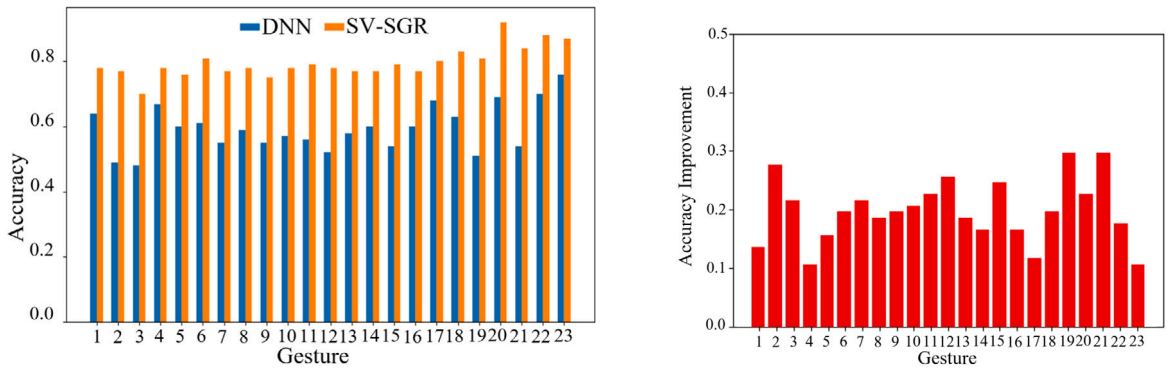
Based on the feedback optimization, the average accuracy of recognizing similar gestures increases from 59.39% to 64.30%, resulting in an improvement of 4.91% (Fig. 11). However, the recognition accuracy of gestures 1, 5, and 13 remains unchanged after feedback enhancement. The recognition accuracy of gestures 16, 17, 18, and 23 exhibits a negative impact. In contrast, the recognition accuracy of gesture 19 increases by 29%, while gesture 20 increases by 13%.

The insignificant improvement in recognition accuracy for similar gestures may be attributed to the low recognition accuracy of the model, leading to inaccurate Shapley value calculation. A model with higher recognition accuracy provides more precise Shapley values. In this regard, increasing the analysis window size to 100×10 has been found to achieve higher recognition accuracy, up to 89%. However, this setting cannot be used in practice due to its large analysis window, which exceeds the 300 ms time limit for the myoelectric interface. Nevertheless, this high-accuracy model can be utilized to calculate Shapley values, which can be applied to the feedback optimization for the model with a 30×10 analysis window. The effectiveness of this approach is demonstrated by the significant improvement in recognition accuracy after feedback enhancement, as illustrated in Fig. 12. The feedback-enhanced model has significantly improved the recognition accuracy of all 23 similar gestures, with an average increase of 20.17%, resulting



(a) Accuracy comparison of 23 similar gestures (b) Accuracy improvement

Fig. 11. Accuracy comparison with a 30×10 analysis window in the DB1 dataset.



(a) Accuracy comparison of 23 similar gestures (b) Accuracy improvement

Fig. 12. Accuracy comparison with a 100×10 analysis window in the DB1 dataset.

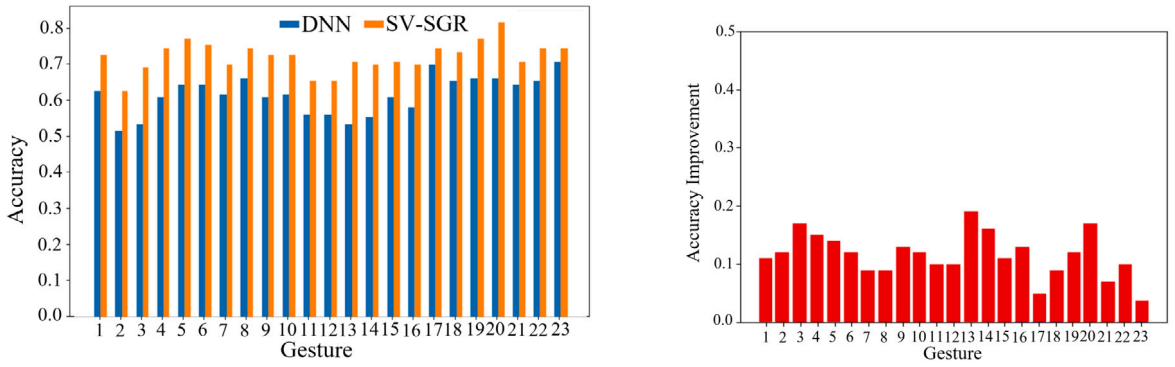
in an overall improvement from 59.39% to 79.56%. The accuracy improvement for gestures 1, 5, and 13 has been particularly noteworthy, with an increase of 14%, 16%, and 19%, respectively. Additionally, the recognition accuracy for gestures 19 and 21 has been enhanced by 30%. These results highlight the significance of utilizing accurate Shapley values in the feedback optimization process to achieve higher recognition accuracy. The use of high-accuracy models for calculating Shapley values effectively improves the overall accuracy of the feedback-enhanced model, indicating its potential for practical applications in myoelectric interface systems.

4.3.4. Accuracy improvement in DB2 and DB3 datasets

We conducted feedback optimization on the recognition model in the DB2 dataset using the same experimental method used in the DB1 dataset. By utilizing the Shapley value based on a 400×12 analysis window, we compared the recognition accuracies for the 23 similar gestures before and after feedback enhancement. The comparison results (Fig. 13) demonstrate that the average accuracy improved by 11.61% after feedback enhancement, with the accuracy increasing from 67.87% to 79.48%. SV-SGR exhibits enhanced recognition accuracy for all gestures, especially for gestures 3, 13, and 20, which are improved by 17%, 19%, and 17%, respectively.

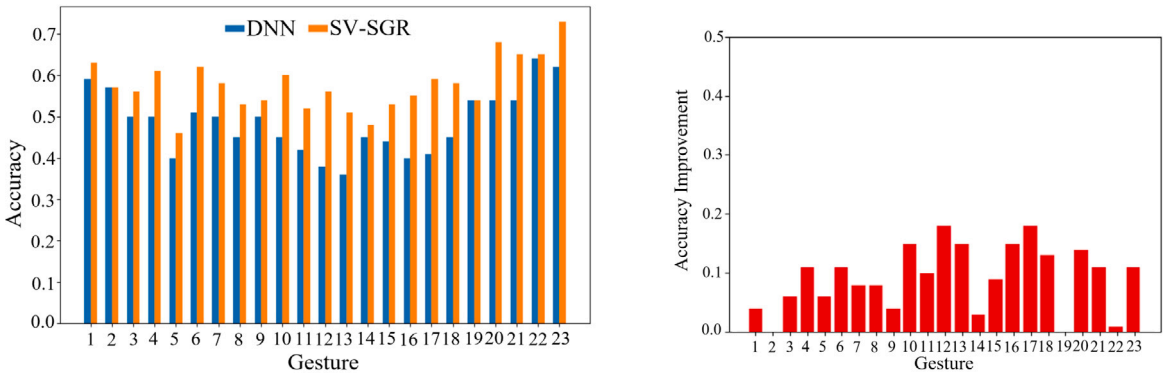
However, the recognition accuracy in the DB3 dataset was unsatisfactory, making it challenging to ensure the accuracy of channel contributions. To address this issue, we enhanced the recognition model by utilizing the Shapley value obtained in the DB2 dataset with the same analysis window size of 400×12. We then compared the accuracy of recognizing 23 similar gestures before and after feedback enhancement in the DB3 dataset, as depicted in Fig. 14(a), and demonstrated the accuracy improvement of each gesture after feedback optimization [Fig. 14(b)].

The experimental results indicate that the recognition performance has been moderately improved, but the recognition accuracy after feedback enhancement is comparatively lower than the results obtained in the DB2 dataset. Specifically, the average accuracy for 23 similar gestures has improved by 9.17%, with the accuracy increasing from 48.52% to 57.69%. While the recognition accuracy can not be improved for gestures 2 and 19, the recognition accuracy for other gestures has been enhanced, particularly for gestures 12 and 17, which improved by 18%. It is noteworthy that all experimental conditions were identical between the DB2 and DB3 datasets, except for the difference in participants. The observed variances in the experimental results can be attributed to individual differences,



(a) Accuracy comparison of 23 similar gestures (b) Accuracy improvement

Fig. 13. Accuracy comparison with a 400×12 analysis window in the DB2 dataset.



(a) Accuracy comparison of 23 similar gestures (b) Accuracy improvement

Fig. 14. Accuracy comparison with a 400×12 analysis window in the DB3 dataset.

where the DB2 dataset consists of a healthy population, while the DB3 dataset comprises amputees. This finding emphasizes the significant influence of individual characteristics on the channel contributions computed using Shapley values.

5. Discussion

This study offers valuable insights into the field of sEMG-based similar gesture recognition. It provides a practical solution that combines deep learning and game theory to achieve high recognition accuracy and interpretability. Rather than utilizing complex deep learning models, a straightforward DNN was utilized as the original recognition model, and combined with feedback optimization techniques, highlighting the contribution of sEMG electrode channels in the recognition process. The experimental results demonstrate that improving the accuracy of Shapley values can obtain more accurate channel contributions and enhance the recognition model through feedback optimization, thus enhancing the performance of recognizing similar gestures. Specifically, in the DB1 dataset, the feedback enhancement results in an average accuracy improvement from 59.39% to 79.56%, while in the DB2 dataset, the accuracy is improved from 67.87% to 79.48%. Moreover, the interpretable analysis results demonstrate that the contribution of electrode channels in recognizing similar gestures exhibits variability across different gestures, which can be leveraged as a distinguishing feature in the recognition process.

However, one limitation of this study, as demonstrated in the DB3 dataset, is that the precise calculation of Shapley values depends on achieving accurate recognition during model training and testing. In addition, individual characteristics may impact the analysis of Shapley values, and the improvement of recognition accuracy may vary based on individual differences. Moreover, a noteworthy finding is that enhancing the recognition model for a specific gesture can also yield improvements in recognizing other similar gestures, which warrants further exploration in future research.

In practical applications, the Shapley value offers the capability to explain the recognition model, which helps users understand the model's decision-making process. The methodology devised in this study aids users in understanding the mechanism behind recognizing similar gestures by visualizing the contributions of electrode channels and extracting features based on the variations in channel contributions to enhance the performance of gesture recognition. Moreover, feedback optimization based on Shapley values facilitates users in rapidly optimizing the recognition model and improving the recognition performance. Overall, the explainable

analysis method based on Shapley values effectively enhances the performance of recognizing similar gestures and greatly assists in elucidating the decision-making process of the recognition model.

6. Conclusion

This study offers a fresh perspective on how to recognize similar gestures using sEMG signals. Unlike conventional methods, we have developed a Shapley-value-based solution for similar gesture recognition, which emphasizes the contribution of sEMG electrode channels in the recognition process. Our global explanation approach utilizes Shapley values to quantify the contribution of each channel and provides a solid foundation for recognizing similar gestures. This explanation analysis enhances the understanding of the recognition process and provides feedback on the model to improve its performance. To evaluate our method, we conducted various comparisons on real-world datasets. The experimental results demonstrate that our SV-SGR method performs better than other baselines and validates the crucial role of channel contributions in recognizing similar gestures. Moreover, our method is more practical than other baselines in real-world applications.

Calculating Shapley values for many players (channels in our case) may result in poor computational efficiency. Investigating this problem in the context of gesture recognition is theoretically meaningful and will be a possible future direction of our research. We also found that individual differences can result in poor recognition performance, particularly for amputees, and we plan to devise a transfer learning method to recognize prosthetic hand movements for amputees in future work. For the loss of temporal information due to image conversion of sEMG signals, we plan to explore a time-to-space mapping approach whereby temporal information will be encoded with the spatial information of the image. Moreover, our study demonstrates that the Shapley-value-based recognition method is practical in myoelectric human-robot interaction and can be applied to robotic prostheses, hand exoskeletons, and many other rehabilitation and assistive devices.

CRediT authorship contribution statement

Feng Wang: Writing – review & editing, Writing – original draft, Methodology. **Xiaohu Ao:** Writing – review & editing, Writing – original draft, Investigation. **Min Wu:** Supervision. **Seiichi Kawata:** Validation, Supervision. **Jinhua She:** Writing – review & editing, Supervision, Methodology.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Grants 62106240 and 61873348; China Postdoctoral Science Foundation under Grant 2022M722943; the Natural Science Foundation of Hubei Province, China, under Grant 2020CFA031; Wuhan Applied Foundational Frontier Project under Grant 2020010601012175; the 111 Project under Grant B17040; and JSPS (Japan Society for the Promotion of Science) KAKENHI under Grants 23K25252 and 22H03998.

References

[1] P. Angelov, E. Soares, R. Jiang, N. Arnold, P. Atkinson, Explainable artificial intelligence: an analytical review, Wiley Interdiscip. Rev. Data Min. Knowl. Discov. 11 (5) (2021) e1424.
[2] M. Atzori, A. Gijsberts, C. Castellini, B. Caputo, A.M. Hager, S. Elsig, G. Giatsidis, F. Bassetto, H. Müller, Electromyography data for non-invasive naturally-controlled robotic hand prostheses, Sci. Data 1 (2014) 140053.
[3] I. Batzianoulis, S. El-Khoury, E. Pirondini, M. Coscia, S. Micera, A. Billard, EMG-based decoding of grasp gestures in reaching-to-grasping motions, Robot. Auton. Syst. 91 (2017) 59–70.
[4] F.S. Botros, A. Phinyomark, E.J. Scheme, Electromyography-based gesture recognition: is it time to change focus from the forearm to the wrist?, IEEE Trans. Ind. Inform. 18 (1) (2022) 174–184.
[5] B. Chakraborty, S. Mukhopadhyay, Heterogeneous recurrent spiking neural network for spatio-temporal classification, Front. Neurosci. (2023), https://doi.org/10.3389/fnins.2023.994517.
[6] H. Chen, I. Covert, S. Lundberg, S. Lee, Algorithms to estimate Shapley value feature attributions, Nat. Mach. Intell. 5 (2023) 590–601.
[7] L. Chen, Y. Li, Y. Liu, Human body gesture recognition method based on deep learning, in: Proceedings of the 2020 Chinese Control and Decision Conference (CCDC 2020), 2020, pp. 587–591.
[8] C. Ding, S. Wen, W. Ding, K. Liu, E. Belyaev, Temporal segment graph convolutional networks for skeleton-based action recognition, Eng. Appl. Artif. Intell. 110 (2022) 104675.
[9] W. Geng, Y. Du, W. Jin, W. Wei, Y. Hu, J. Li, Gesture recognition by instantaneous surface EMG images, Sci. Rep. 6 (2016) 36571.
[10] P. Gulati, Q. Hu, S.F. Atashzar, Toward deep generalization of peripheral EMG-based human-robot interfacing: a hybrid explainable solution for neuroRobotic systems, IEEE Robot. Autom. Lett. 6 (2) (2021) 2650–2657.

[11] D. Gunning, M. Stefik, J. Choi, T. Miller, S. Stumpf, G. Yang, XAI—explainable artificial intelligence, Sci. Robot. 4 (37) (2019) eaay7120.
[12] Y. Huang, J. Huang, X. Wu, Y. Jia, Dynamic sign language recognition based on CBAM with autoencoder time series neural network, Mob. Inf. Syst. (2022) 3247781.
[13] A. Kashizadeh, K. Peñan, A. Belford, A. Razmjou, M. Asadnia, Myoelectric control of a biomimetic robotic hand using deep learning artificial neural network for gesture classification, IEEE Sens. J. 22 (19) (2022) 18914–18921.
[14] W. Ke, Y. Xing, C.G. Di Caterina, L. Petropoulakis, J. Soraghan, Intersected EMG heatmaps and deep learning based gesture recognition, in: Proceedings of the 12th International Conference on Machine Learning and Computing (ICMLC 2020), 2020, pp. 73–78.
[15] W. Li, P. Shi, H. Yu, Gesture recognition using surface electromyography and deep learning for prostheses hand: state-of-the-art, challenges, and future, Front. Neurosci. 15 (2021) 621885.
[16] Z. Li, X. Zhao, Z. Wang, R. Xu, L. Meng, D. Ming, A hierarchical classification of gestures under two force levels based on muscle synergy, Biomed. Signal Process. Control 77 (2022) 103695.
[17] X. Luo, X. Wu, L. Chen, Y. Zhao, L. Zhang, G. Li, W. Hou, Synergistic myoelectrical activities of forearm muscles improving robust recognition of multi-fingered gestures, Sensors 19 (3) (2019) 610.
[18] P. Molchanov, S. Gupta, K. Kim, J. Kautz, Hand gesture recognition with 3D convolutional neural networks, in: Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW 2015), 2015.
[19] M.A. Ozdemir, D.H. Kisa, O. Guren, A. Akan, Dataset for multi-channel surface electromyography (sEMG) signals of hand gestures, Data Brief 41 (2022) 107921.
[20] S. Pancholi, A.M. Joshi, D. Joshi, A robust and accurate deep learning based pattern recognition framework for upper limb prosthesis using sEMG, arXiv, Signal Process. (2021), https://doi.org/10.48550/arXiv.2106.02463.
[21] K. Ren, T. Zheng, Z. Qin, X. Liu, Adversarial attacks and defenses in deep learning, Engineering 6 (3) (2020) 346–360.
[22] F.M. Siemers, J. Bajorath, Differences in learning characteristics between support vector machine and random forest models for compound classification revealed by Shapley value analysis, Sci. Rep. 13 (2023) 5983.
[23] Y. Sun, Y. Weng, B. Luo, G. Li, B. Tao, D. Jiang, D. Chen, Gesture recognition algorithm based on multi-scale feature fusion in RGB-D images, IET Image Process. 17 (2023) 1280–1290.
[24] S. Tam, M. Boukadoum, A. Campeau-Lecours, B. Gosselin, A fully embedded adaptive real-time hand gesture classifier leveraging HD-sEMG and deep learning, IEEE Trans. Biomed. Circuits Syst. 14 (2019) 232–243.
[25] X. Tang, Z. Yan, J. Peng, B. Hao, H. Wang, J. Li, Selective spatiotemporal features learning for dynamic gesture recognition, Expert Syst. Appl. 169 (2021) 114499.
[26] G. Tong, Y. Li, H. Zhang, N. Xiong, A fine-grained channel state information-based deep learning system for dynamic gesture recognition, Inf. Sci. 636 (2023) 118912.
[27] M.C. Tosin, J.C. Machado, A. Balbinot, sEMG-based upper limb movement classifier: current scenario and upcoming challenges, J. Artif. Intell. Res. 75 (2022) 83–127.
[28] S. Wang, A. Wang, M. Ran, L. Liu, Y. Peng, M. Liu, G. Su, A. Alhudhaif, F. Alenezi, N. Alnaim, Hand gesture recognition framework using a Lie group based spatio-temporal recurrent network with multiple hand-worn motion sensors, Inf. Sci. 606 (2022) 722–741.
[29] D.S. Watson, J. O'Hara, N. Tax, R. Mudd, I. Guy, Explaining predictive uncertainty with information theoretic Shapley values, eprint arXiv:2306.05724, https://arxiv.org/abs/2306.05724, 2023.
[30] W. Yang, Y. Wei, H. Wei, Y. Chen, G. Huang, X. Li, R. Li, N. Yao, X. Wang, X. Gu, M.B. Amin, B. Kang, Survey on explainable AI: from approaches, limitations and applications aspects, in: Human-Centric Intelligent Systems, 2023.
[31] S. Zabihi, E. Rahimian, A. Asif, A. Mohammadi, TraHGR: transformer for hand gesture recognition via electroMyography, arXiv, Signal Process. (2022), https://doi.org/10.48550/arXiv.2203.16336.
[32] K. Zeissler, Gesture recognition gets an update, Nat. Electron. 6 (2023) 272.
[33] H. Zhang, L. Wang, Y.Q. Wang, Spatio-temporal attention-based deep neural networks for action recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017), 2017, pp. 6450–6459.
[34] J. Zhang, L. Wang, Q. Liu, Saliency-enhanced deep feature-based approach for first-person action recognition, IEEE Trans. Circuits Syst. Video Technol. 29 (2) (2018) 420–430.
[35] Y. Zhang, Y. Liao, X. Wu, L. Chen, Q. Xiong, Z. Gao, X. Zheng, G. Li, W. Hou, Non-uniform sample assignment in training set improving recognition of hand gestures dominated with similar muscle activities, Front. Neurorobot. 12 (2018) 3.
[36] M. Zheng, M.S. Crouch, M.S. Eggleston, Surface electromyography as a natural human-machine interface: a review, IEEE Sens. J. 22 (10) (2022) 9198–9214.
[37] Y. Zheng, L. Xiao, Classifying object of standard grasping movements using data glove with LSTM networks, in: Proceedings of the 7th International Conference on Computer and Communications (ICCC 2021), 2021, pp. 831–835.
[38] T. Zhou, O.M. Omisore, W. Du, L. Wang, Y. Zhang, Adapting random forest classifier based on single and multiple features for surface electromyography signal recognition, in: Proceedings of the 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI 2019), 2019, pp. 1–6.
[39] J.L. Betthauser, J.T. Krall, S.G. Bannowsky, G. Levay, R.R. kaliki, M.S. Fifer, N.V. Thakor, Stable responsive EMG sequence prediction and adaptive reinforcement with temporal convolutional networks, IEEE Trans. Biomed. Eng. 67 (6) (2020) 1707–1717.
[40] M. Ergeneci, E. Bayram, D. Binningsley, D. Carter, P. Kosmas, Attention-enhanced frequency-split convolution block for sEMG motion classification: experiments on premier league and NinaPro datasets, IEEE Sens. J. 24 (4) (2024) 4821–4830.
[41] N.K. Karnam, S.R. Dubey, A.C. Turlapaty, B. Gokaraju, EMGHandNet: a hybrid CNN and Bi-LSTM architecture for hand activity classification using surface EMG signals, Biocybern. Biomed. Eng. 42 (1) (2022) 325–340.
[42] E. Rahimian, S. Zabihi, A. Asif, D. Farina, S.F. Atashzar, A. Mohammadi, FS-HGR: few-shot learning for hand gesture recognition via electromyography, IEEE Trans. Neural Syst. Rehabil. Eng. 29 (2021) 1004–1015.
[43] R.V. Godoy, A. Dwivedi, M. Liarokapis, Electromyography based decoding of dexterous, in-hand manipulation motions with temporal multichannel vision transformers, IEEE Trans. Neural Syst. Rehabil. Eng. 30 (2022) 2207–2216.
[44] M.D. Dere, B. Lee, A novel approach to surface EMG-based gesture classification using a vision transformer integrated with convolutive blind source separation, IEEE J. Biomed. Health Inform. 28 (1) (2024) 181–192.
[45] R. Azad, M. Asadi-Aghbolaghi, S. Kasaei, S. Escalera, Dynamic 3D hand gesture recognition by learning weighted depth motion maps, IEEE Trans. Circuits Syst. Video Technol. 29 (6) (2019) 1729–1740.
[46] T. Feix, J. Romero, H.B. Schmiedmayer, A.M. Dollar, D. Kragic, The GRASP taxonomy of human grasp types, IEEE Trans. Human-Mach. Syst. 46 (1) (2016) 66–77.
[47] A. Calado, P. Roselli, V. Errico, N. Magrofuoco, J. Vanderdonckt, G. Saggio, A geometric model-based approach to hand gesture recognition, IEEE Trans. Syst. Man Cybern. Syst. 52 (10) (2022) 6151–6161.