TYPE Original Research
PUBLISHED 28 February 2023
DOI 10.3389/fnbot.2023.1127338

LST-EMG-Net: Long short-term transformer feature fusion network for sEMG gesture recognition

Wenli Zhang[1]*, Tingsong Zhao[1], Jianyi Zhang[2] and Yufei Wang[1]

[1]Faculty of Information Technology, Beijing University of Technology, Beijing, China, [2]College of Art and Design, Beijing University of Technology, Beijing, China

OPEN ACCESS

EDITED BY
R. A. R. C. Gopura,
University of Moratuwa, Sri Lanka

REVIEWED BY
Asif Ali Laghari,
Sindh Madressatul Islam University, Pakistan
Yang Sun,
Shenyang Normal University, China

*CORRESPONDENCE
Wenli Zhang
✉ zhangwenli@bjut.edu.cn

RECEIVED 19 December 2022
ACCEPTED 14 February 2023
PUBLISHED 28 February 2023

CITATION
Zhang W, Zhao T, Zhang J and Wang Y (2023) LST-EMG-Net: Long short-term transformer feature fusion network for sEMG gesture recognition.
Front. Neurorobot. 17:1127338.
doi: 10.3389/fnbot.2023.1127338

With the development of signal analysis technology and artificial intelligence, surface electromyography (sEMG) signal gesture recognition is widely used in rehabilitation therapy, human-computer interaction, and other fields. Deep learning has gradually become the mainstream technology for gesture recognition. It is necessary to consider the characteristics of the surface EMG signal when constructing the deep learning model. The surface electromyography signal is an information carrier that can reflect neuromuscular activity. Under the same circumstances, a longer signal segment contains more information about muscle activity, and a shorter segment contains less information about muscle activity. Thus, signals with longer segments are suitable for recognizing gestures that mobilize complex muscle activity, and signals with shorter segments are suitable for recognizing gestures that mobilize simple muscle activity. However, current deep learning models usually extract features from single-length signal segments. This can easily cause a mismatch between the amount of information in the features and the information needed to recognize gestures, which is not conducive to improving the accuracy and stability of recognition. Therefore, in this article, we develop a long short-term transformer feature fusion network (referred to as LST-EMG-Net) that considers the differences in the timing lengths of EMG segments required for the recognition of different gestures. LST-EMG-Net imports multichannel sEMG datasets into a long short-term encoder. The encoder extracts the sEMG signals' long short-term features. Finally, we successfully fuse the features using a feature cross-attention module and output the gesture category. We evaluated LST-EMG-Net on multiple datasets based on sparse channels and high density. It reached 81.47, 88.24, and 98.95% accuracy on Ninapro DB2E2, DB5E3 partial gesture, and CapgMyo DB-c, respectively. Following the experiment, we demonstrated that LST-EMG-Net could increase the accuracy and stability of various gesture identification and recognition tasks better than existing networks.

KEYWORDS

sEMG signals, gesture recognition, multi-scale features, multi-head attention, stroke rehabilitation, human-computer interaction

1. Introduction

Surface electromyography signals are bioelectric signals generated during muscle contractions. sEMG signals can be collected non-invasively, safely, and easily, and sEMG can directly reflect the state of muscle activity. By analyzing the sEMG, gestures can be accurately recognized. sEMG-based gesture recognition methods have the advantages of being faster and more environmentally independent than vision-based gesture recognition methods (Oudah et al., 2020; Mujahid et al., 2021). Therefore, sEMG-based gesture recognition methods have strong application possibilities in sectors related to human-computer interfaces, including intelligent prosthetics (Cipriani et al., 2008), upper limb rehabilitation exoskeletons (Leonardis et al., 2015), robotic arm control (Wang et al., 2012), and others (Muri et al., 2013).

An sEMG-based gesture recognition framework generally consists of four parts: Signal preprocessing, signal segmentation, feature extraction, and gesture classification mechanisms (Parajuli et al., 2019). For traditional machine learning algorithms, the features used for classification are usually handcrafted by human experts. Therefore, the quality of the feature set selected by the experts directly determines the success or failure of the recognition task. Numerous gesture recognition studies have used traditional classifiers for manual features. For example, support vector machines (SVMs) (Alseed and Tasoglu, 2022; Briouza et al., 2022), k-nearest neighbors (KNN) (Baygin et al., 2022), linear discriminant analysis (LDA) (Narayan, 2021), hidden Markov models (HMMs) (Hu and Wang, 2020), and random forests (RF) (Xue et al., 2019; Jia et al., 2021) have made some progress. However, the accuracy and stability of traditional learning algorithms do not yet satisfy practical application requirements when applied to large-scale datasets consisting of larger numbers of hand gestures or wrist movements. Therefore, improving the accuracy and stability of hand gesture recognition is still a hot research topic.

In recent years, with the rapid development of artificial intelligence technology, deep learning has shown great potential in medical rehabilitation fields such as physiological signal processing (Rim et al., 2020; Al-Saegh et al., 2021) and medical image imaging (Karim et al., 2022; Laghari et al., 2022). In gesture recognition tasks based on surface EMG signals, deep learning methods can automatically learn the dependencies or intrinsic connections of the amplitude changes at each sampling point in surface EMG signals due to their deep network architectures. The dependencies and intrinsic connections can be considered the muscle activity features that indirectly express forearm muscle activity conditions, and gesture information can be obtained under this condition. The following research summarizes the feature extraction methods that have been developed under different model architectures for deep learning algorithms.

1.1. Related work

Deep learning models outperform traditional machine learning models, so many researchers use deep learning for gesture recognition. Convolutional neural networks (CNNs) (Atzori et al., 2016; Wei et al., 2019; Chen et al., 2020), recurrent neural networks (RNNs) (Vaswani et al., 2017; Hu et al., 2018; Xia et al., 2018), and transformer-based gesture identification approaches (Rahimian et al., 2021; Siddhad et al., 2022) are the current prevalent deep learning gesture recognition algorithms.

Researchers have conducted studies on CNN-based gesture recognition methods (Atzori et al., 2016; Wei et al., 2019; Chen et al., 2020). Atzori et al. (2016) first applied a CNN to an sEMG gesture recognition task using only a shallow network constructed from four convolutional layers. The accuracy was comparable to that of traditional machine learning gesture recognition methods. Wei et al. (2019) proposed a multistream convolutional neural network (MSCNN) with decomposition and fusion stages. The network learned the correlations between gesture muscles, and it was evaluated on three benchmark databases. The results showed that multistream CNNs outperformed simple CNNs and random forest classifiers, but the computational effort of the method increased multiplicatively with the number of myoelectric channels.

Some researchers have combined recurrent neural networks (RNNs) with CNNs, using CNNs for feature extraction and RNNs for modeling time dependencies (Vaswani et al., 2017; Hu et al., 2018; Xia et al., 2018). An RNN has all nodes connected in a chain-like manner, so it can handle short-term memorization tasks well. For example, Xia et al. (2018) proposed the RCNN, a single-branch deep structure with a CNN and RNN connected serially. The CNN extracts the myoelectric local spatial features, and the RNN saves the local spatial features and efficiently passes them to the next moment to update the model weights. This network has an advantage in learning complex motion features. However, the RCNN can have a sharp decrease in recognition accuracy over time compared to the CNN. Xia et al. (2018) tried to use large neural networks with more layers and parameters to improve the robustness of the model to time variations. Nevertheless, problems such as a heavy training time burden and system recognition delays are caused by the inability of RNNs to train in parallel. Hu et al. (2018) proposed an attention-based hybrid CNN-RNN model. The model uses a CNN to extract spatial feature maps of successive frames of sEMG signals and an RNN to further extract temporal features from the feature maps. The model was able to effectively extract the temporal correlation of each channel of sparse multichannel sEMG signals.

In recent years, after the transformer model (Vaswani et al., 2017) was proposed, it attracted attention in natural language processing and computer vision tasks. The transformer model entirely relies on self-attention, which can capture global dependencies in the input data to achieve parallel computation and improve computational efficiency. At the same time, self-attention can produce more interpretable models. For example, Siddhad et al. (2022) explored the effectiveness of transformer networks for the classification of raw EEG data. They used raw resting-state EEG data to classify people by age and gender, and the classification results showed that the transformer network was comparable in accuracy to state-of-the-art CNN and LSTM recognition with feature extraction. This proved that the transformer network has excellent feature extraction capability for time-series signals. Some researchers have already used transformers for hand gesture recognition (Rahimian et al., 2021; Montazerin et al., 2022). For example, Rahimian et al. (2021) used a vision-based transformer model architecture for the classification and recognition of upper limb gestures, and the recognition accuracy on Ninapro DB2 Exercise B for 17 gestures reached state-of-the-art performance at that time. Montazerin et al. (2022) also proposed a transformer framework based on ViT for high-density sEMG gesture recognition with 128 arrayed electrodes, which can solve the time burden problem of RNN structures that are not trained in parallel. However, current transformer-based gesture recognition networks only apply the image classification scheme to EMG recognition. The network structure is not designed according to the characteristics of gesture activity and sEMG signals.

In summary, gesture recognition is mainly implemented by deep learning methods at this stage. Among them, the transformer model has become a hot research topic because of its self-attention structure, which can extract sEMG signal temporal muscle activity features well and performs well in gesture recognition. However, current recognition methods still suffer from a mismatch between the amount of information contained in the extracted features and the amount of information required to recognize gestures when implementing multicategory gesture recognition. The reason for the mismatch problem is that there are differences in the stability exhibited in the sEMG signal due to the different muscle activity, muscle contraction changes, and muscle strength changes mobilized (Farago et al., 2022; Li et al., 2022). The sEMG signals of more complex gestures are less stable, and simpler gesture movements have better EMG signal stability. To recognize complex gestures from less stable EMG signals, the lengths of the feature extraction segments need to increase to yield a sufficient amount of information for recognition (Nazmi et al., 2017). However, most of the existing related works do not consider the characteristic that the lengths of EMG signals are different for different gestures. They all intercept fixed-length EMG signals for spatial and temporal feature extraction, which leads to a mismatch between the amount of feature information extracted by the designed models and the corresponding gestures and affects the accuracy and robustness of the gesture recognition framework.

1.2. Contributions

To address the above problems, it is necessary to propose a gesture recognition method to extract moderate feature information from EMG sample segments. Therefore, we propose a gesture recognition method based on LST-EMG-Net. It can extract long- and short-sequence features in sEMG windows and fuse them effectively to achieve high-accuracy recognition of complex and simple gestures. The method proposed in this article makes the following three contributions:

(1) To address the mismatch between the feature information and required information in a multicategory gesture recognition task, we propose a long short-term encoder and use the linear projection in the encoder to construct a long-term branch and a short-term branch. Then, each branch feature is extracted by a long- or short-term subencoder to achieve multiscale time feature extraction and solve the problem of redundant or insufficient feature extraction. To further improve the feature quality, we use sEMG channel attention to dynamically set the weights of each channel of the sEMG windows.

(2) We propose a cross-attention module for long- and short-term features from the encoder to fuse the long- and short-term features efficiently. This module uses an attention-based approach to cross-learn one branch's classification token and another branch's patch tokens in the feature. This module can effectively fuse the muscle activity information and enhance the efficiency of feature fusion due to its low computational effort. It finally achieves the goal of improving the accuracy of hand gesture recognition.

(3) To address the problem that individual sEMG signals are difficult to collect in large quantities, we propose a signal augmentation method based on sEMG windows. This method adopts random windows and sEMG time delays to augment the original sEMG windows and constructs a training dataset together with the original EMG timing windows. This method reduces the burden of data collection.

The remainder of the article is organized as follows. The dataset and the sEMG signal enhancement method utilized in this article are described in detail in Section "2. Materials and methods." The framework of LST-EMG-Net, including the motivation of the study and the submodule structure, is presented in Section "3. The long short-term sEMG transformer feature fusion network framework." The experimental environment of LST-EMG-Net and experimental results are presented in Section "4. Experiments and results." Finally, the conclusions of this article are drawn in Section "5. Conclusion."

2. Materials and methods

2.1. The datasets

We use two types of datasets, a sparse sEMG dataset and a high-density sEMG dataset, to evaluate our LST-EMG-Net. The sparse dataset includes Ninapro DB2 and DB5 (Atzori et al., 2012, 2014a,b; Gijsberts et al., 2014; Du et al., 2017; Pizzolato et al., 2017). The high-density dataset is the public CapgMyo dataset (Du et al., 2017).

Sparse sEMG dataset: We use 17 basic wrist movements and isotonic hand configurations from the DB2 Exercise B subdataset (as shown in **Figure 1A**). In the DB2 dataset, the muscular activities were measured using 12 active double-differential wireless electrodes from a Delsys Trigno Wireless EMG system at a sampling frequency of 2 kHz. The DB5 dataset uses 18 gestures from the Exercise C subdataset that fully mobilize muscle activity and facilitate muscle recovery training (as shown in **Figure 1B**). The DB5 dataset was taken from 10 healthy subjects. Its collection device was a pair of Thalmic Labs Myo (Myo) armbands. Each Myo had eight single-channel electrodes, each with a sampling rate of 200 Hz. The DB2 dataset and DB5 dataset collection rules were the same. Each gesture was repeated six times, each acquisition had a

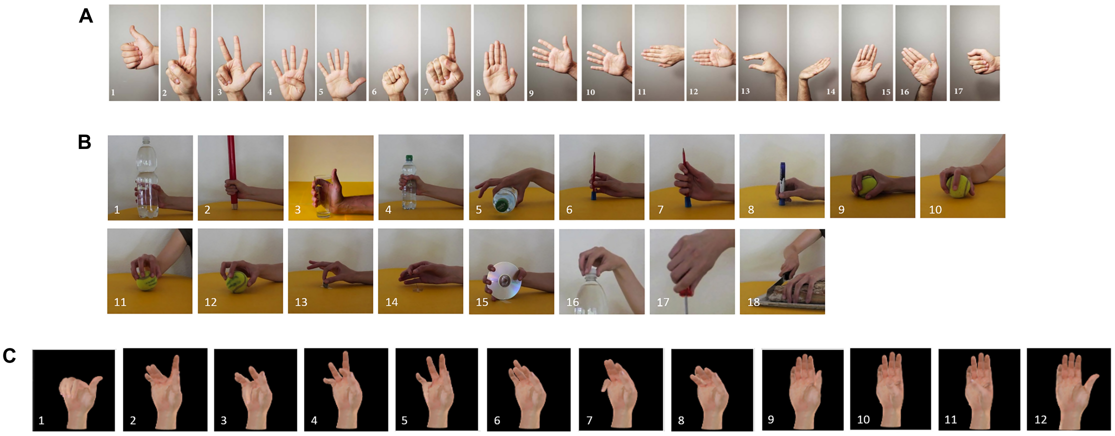

FIGURE 1
Types of gestures in the datasets used in this manuscript. **(A)** Ninapro DB2 exercise B dataset 17 gestures. **(B)** Ninapro DB5 exercise C dataset 18 gestures. **(C)** CapgMyo DB-c dataset 12 gestures.

5 s activity signal, and each acquisition interval was 3 s. The 1st, 3rd, 4th, and 6th repetitions of the gesture were used to construct the training set, and the 2nd and 5th repetitions were used to build the test set.

High-density sEMG dataset: We used 12 basic finger movements from the DB-C subdataset of CapgMyo (as shown in **Figure 1C**). The dataset was acquired at a sampling rate of 1,000 Hz with an array of 8 2 × 8 differential electrodes to capture the activity of the right peripheral forearm muscle groups. The CapgMyo dataset was obtained from 10 users who repeated several movements 10 times, each lasting 3 s, followed by 7 s of rest. Odd-numbered repetitions were used to construct the training set, and even-numbered repetitions were used to build the test set.

2.2. Preprocessing

Before performing the classification task, the sEMG signals were preprocessed. The sEMG signals were filtered from power line interference before signal acquisition due to the built-in 50 Hz trap filter in the sEMG sensor. We only used a blind source separation process called fast independent component analysis (Fast ICA) (Comon, 1992) on the raw signals to eliminate interchannel crosstalk. Then, Z Score standard normalization was used to process the sEMG signals after filtering the noise. Z Score normalization of a channel is shown in Equation 1.

$$F(x_t) = \frac{x_t - \mu}{\sigma} \tag{1}$$

Where x_t is the sEMG signal, μ is the mean value of the sEMG signal and σ is the standard deviation of the sEMG signal.

This article uses the sliding-window method with overlap to segment the normalized EMG signal to obtain the original EMG timing window. The length of the sliding window is set according to the related work of Scheme and Englehart (2011). It is noted that 300–800 ms is an acceptable latency. Considering the delay and computation volume, we set the window length of the Ninapro DB2 dataset as 300 ms, its window distance as 10 ms, the window length of the Ninapro DB5 dataset as 500 ms, its window length as 100 ms, the window length of the CapgMyo dataset as 60 ms and its window distance as 10 ms.

2.3. Signal augmentation based on sEMG windows (SA)

Due to the lack of *a priori* experience of the self-attention of the transformer network, such as inductive bias and translational invariance, the transformer model requires a larger dataset to reach convergence. However, most current recognition methods require the acquisition of individual sEMG signals to build recognition models, whereas the collection of sEMG data from a large number of individuals is difficult to achieve because of the high time cost and muscle fatigue. Therefore, we propose a signal augmentation method based on sEMG windows to solve the above problem. This method is used to obtain random windows and time-delay enhancement windows to increase the number of training samples.

First, the original sEMG signals are input into the random window selection module. This module randomly selects the start point of the window within each class of gesture action sequences and determines the end point based on the window length to obtain a random window for that type of gesture. This operation is repeated to obtain the random windows for all gesture actions.

Second, the transceiver delay and transmission interference of the acquisition device (Liu et al., 2016) make it inevitable that sEMG will miss some sample points, which impacts the sEMG recognition model's robustness. Therefore, this step randomly selects a certain percentage of the original sEMG window to input to the time-delay enhancement module. This model selects sequence An in the original window randomly and selects sequence B at the next sampling moment (where the numbers of sampling points of sequence A and sequence B are the same); finally, sequence A is deleted, and the sampling points of sequence B are copied to the original sequence A sampling moment to obtain the time-delay enhancement windows, as shown in **Figure 2**.

The proposed signal augmentation method expands the training samples by doubling the number of sEMG windows. Take

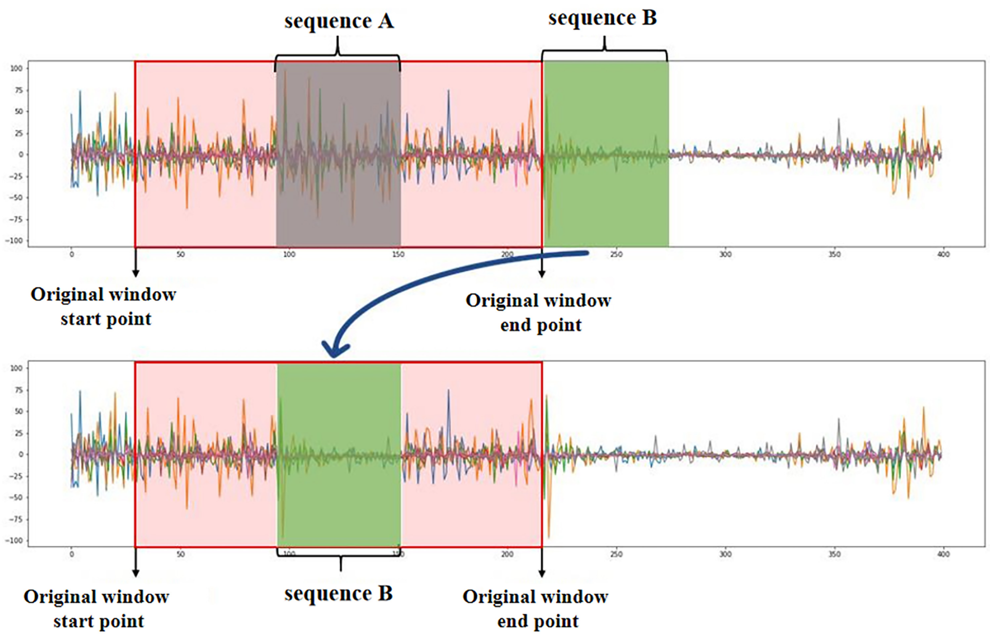
sequence A
sequence B
100
75
50
25
0
−25
−50
−75
−100
0
50
100
150
200
250
300
350
400
Original window start point
Original window end point
sequence B
Original window start point
Original window end point

FIGURE 2
Schematic diagram of time-delay enhancement module.

DB5 dataset subject ten as an example: we have 5,886 original EMG windows initially; then, we obtain 2,943 random windows by the random window selection module and 2,943 time-delay enhanced windows by the time-delay enhancement module. Finally, we obtain 11,772 windows.

3. The long short-term sEMG transformer feature fusion network framework

In the field of gesture recognition, we often use the fractal dimension to calculate the complexity and stability of the EMG signal (Naik and Nguyen, 2014). Gestures with low fractal dimensions are simple gestures whose signal stability is higher, and the electrode position, muscle contraction, and muscle force change more slowly in these types of gestures; they include single-finger flexion, multifinger flexion, and wrist translation (Namazi, 2019a). Gestures with high fractal dimensions are complex gestures with low signal smoothness. The electrode position, muscle contraction, and muscle force change more rapidly in these gestures, such as wrist-hand linkage and dynamic operations (e.g., grasping, pressing, and tapping). In addition, when the subject increases the force of the gesture, it also leads to an increase in the fractal dimension of the EMG signal (Menon et al., 2017; Namazi, 2019b), which in turn affects the stability of the EMG signal. Therefore, we need longer EMG sequence segments to extract high-quality features from signals with low stability when recognizing complex gestures; we need shorter EMG sequence segments to recognize simple gestures.

On the other hand, the transformer can capture longer dependent information in the temporal signal classification task due to its self-attention structure. However, current transformer networks are designed based on multihead attention; in this method, there is a lack of constraints between every pair of heads, which makes the output similar between network layers, eventually leading to the problem of attention collapse (Zhou et al., 2021) and affecting accuracy. Therefore, we propose LST-EMG-Net to extract the sEMG features of both long-term and short-term segments in the sEMG window to perform multitemporal feature extraction and feature fusion for various-complexity gesture recognition tasks. To further improve the gesture recognition effect, the network adds a transposition matrix between the heads of multihead attention to solve the attention collapse problem.

The overall structure of LST-EMG-Net is shown in **Figure 3**; it consists of three parts: the long short-term encoder, feature cross-attention module, and gesture classification module, of which the long short-term encoder module and the feature cross-attention module correspond to contributions 1 and 2 of this article, respectively.

Long short-term encoder: This module takes as input a set of multichannel sEMG window collections $D = \{(X_i, y_i)\}_{i=1}^{m}$. Then, the input is given importance weights for each channel, and long-term features $\mathrm{Z}_{\mathrm{N}}^{\mathrm{L}}$ and short-term features Z_M^S are extracted from the sEMG window. The temporal window set D consists of m windows; the i window is denoted by $X_i \in \mathbb{R}^{HxW}(1 \leq i \leq M)$, and the gesture label is denoted by y_i. H is the number of EMG signal channels, and W is the number of sampling points per window.

Feature cross-attention module: This module receives the long-term features $\mathrm{Z}_{\mathrm{N}}^{\mathrm{L}}$ and short-term features Z_M^S extracted by the long short-term encoder. The long-term features and short-term features are cross-learned using scaled dot-product attention, and the cross-learned long-term features $\mathrm{Z}_{\mathrm{N}}^{\mathrm{L}'}$ and short-term features $Z_M^{S'}$ are output.

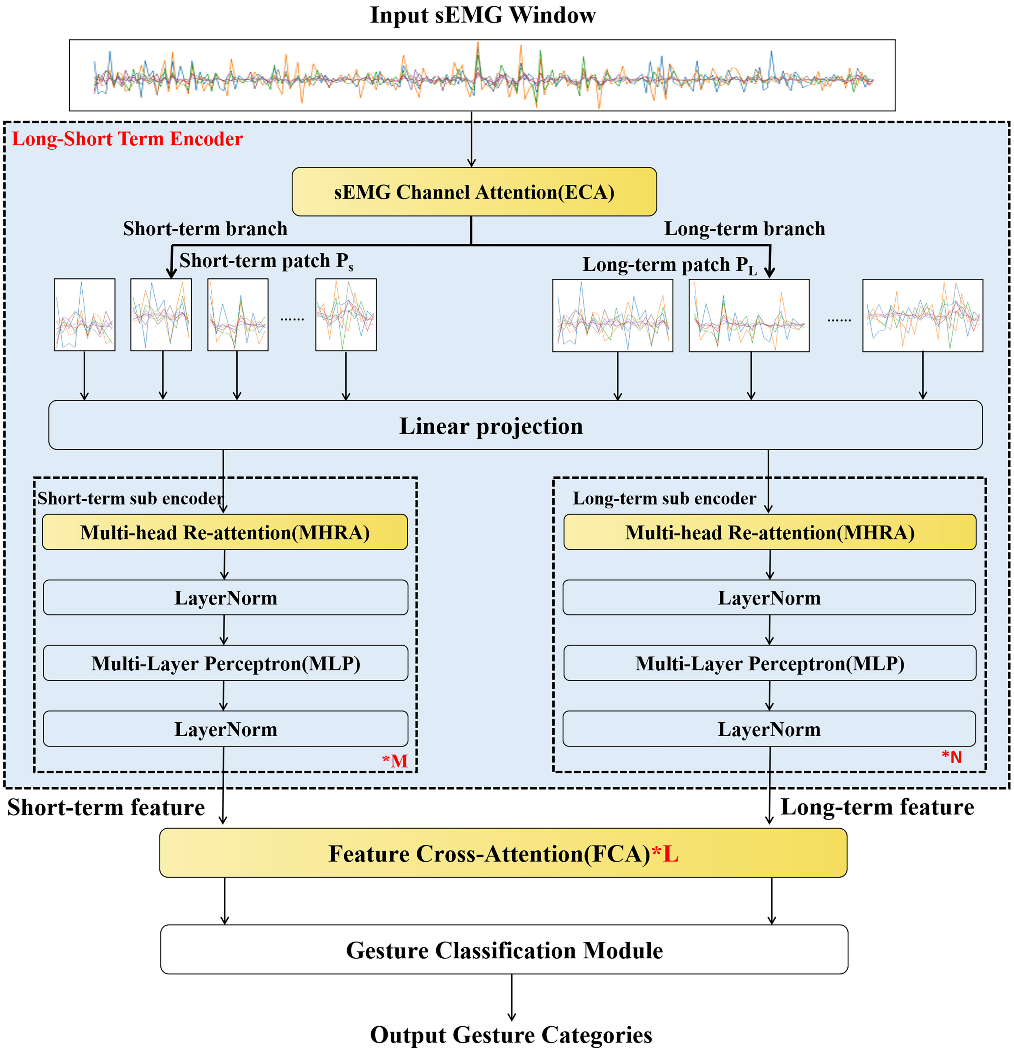

FIGURE 3
The proposed LST-EMG-Net structure. Among them, the sEmg channel attention, multi-head re-attention and the feature cross-attention module (yellow module) are the contribution points of this manuscript.

Gesture classification module: This module receives the long-term features $Z^{L'}_{N}$ and short-term features $Z^{S'}_{M}$ after cross-learning, calculates the gesture category probabilities corresponding to the long short-term features, fuses the gesture probabilities for decision-level fusion and finally outputs the gesture categories.

3.1. Long short-term encoder

This module mainly consists of three parts: sEMG channel attention, linear projection, and the long/short-term sub encoder.

3.1.1. Surface electromyography channel attention (ECA)

It is commonly accepted in medical statistics that sEMG signals from one muscle are statistically independent of those from neighboring muscles (Naik et al., 2007) and that specific muscles play more critical roles in certain hand movements (Huang et al., 2009). However, most of the previous methods extracted correlations between channels and gestures by constructing multistream inputs with channel decomposition signals. As an MSCNN (Wei et al., 2019) assigns network input streams to each channel and fuses them, the computational effort increases exponentially when the number of channels is high. Therefore, to reduce the computational effort, we propose modularized myoelectric channel attention based on scaled dot-product attention to perform correlation extraction of channels and gestures and dynamically adjust the channel weights according to the gestures, increasing the channel weights with strong correlations and decreasing the channel weights with weak correlations.

First, the sEMG window X_i converts each channel into K and Q. Then, we calculate the correlations between channels using scaled point multiplier attention and output the sEMG window with channel weights P_i, as shown in Equation 2.

$$P_i = X_i + Softmax\left(\frac{Avgpooling(Q) \times Avgpooling(K)}{\sqrt{d_k}}\right)X_i \quad (2)$$

where d_{k} is the vector dimension, Avgpooling is the mean pooling layer, $\mathrm{X_i}$ is the raw EMG window, and sotfmax is a normalized exponential function.

3.1.2. Linear projection

To construct long-term and short-term branches for multiscale feature extraction, this module slices P_i into long-term and short-term segments, respectively, and performs linear projection into long-term tokens p_i^L and short-term tokens p_i^S. The final construction forms the set of long-term branching tokens z_0^L. As in Equation 3, the set of short-term branching tokens z_0^S is described in Equation 4.

$$Z_0^S = [p_{\mathrm{cls}}^S; \mathrm{p}_1^S E^S; \mathrm{p}_2^S E^S; \ldots; \mathrm{p}_S^{N_S} E^S] + E_{\mathrm{pos}}^S \tag{3}$$

$$Z_0^L = [p_{\mathrm{cls}}^L; \mathrm{p}_1^L E^L; \mathrm{p}_2^L E^L; \ldots; \mathrm{p}_{N_L}^L E^L] + E_{\mathrm{pos}}^L \tag{4}$$

where E^S and E^L are linear projection matrices, $p_1^s p_2^s \ldots p_{N_S}^s$ are short-term tokens whose sizes are set to $H \times S_{Short}$, $p_1^L p_2^L \ldots p_{N_L}^L$ are long-term tokens whose sizes are set to $H \times S_{Long}$, P_{cls}^S and P_{cls}^L are the classification tokens of the short-term branch and long-term branch, and E_{pos}^S and E_{pos}^L are the position embeddings of the short-term branch and long-term branch, respectively.

3.1.3. Multihead reattention (MHRA)

The long-/short-term subencoder mainly consists of multihead reattention (MHRA) and a multilayer perceptron (MLP). MHRA is the contribution of this module. MHRA collects complementary information about the interactions between multiple attentions by adding a transformation matrix $\theta \in \mathbb{R}^{head \times head}$. MHRA enables individual heads to observe the characteristics of the signal from different angles, effectively solving the attentional collapse problem (Zhou et al., 2021), where head is the number of MHRA output heads.

This module extracts the long- and short-term features from the set of long-term branch tokens z_0^L and short-term branch tokens z_0^S by MHRA and the MLP, respectively. The specific steps of the module are as follows.

First, we compute the interpatch attention information by transforming each patch in the output z_0^S or z_0^L of the linear projection module into QKV, which is fed into the respective branch's encoder.

$$Re\text{−}Attention(Q, K, V) = \mathrm{Norm}(\theta^{\mathrm{T}}(Soft\max(QK^T/\sqrt{d_{\mathrm{k}}})))V \tag{5}$$

where $Norm$ is the layer norm normalization function, Q, K, and V are the query, key and value for the short-term branch, respectively, and d_k is the vector dimension.

Next, the reattention information from the MHRA module is input to the MLP module, and the MHRA and MLP modules are connected by means of residuals.

Finally, the short-term sequence characteristic of the short-term branch output is Z_M^S, as in Equation 6, and similarly, the long-term sequence characteristic Z_N^L can be obtained with Equation 7.

$$Z_M^S = [\mathrm{z}_{\mathrm{cls}}^S; \mathrm{z}_1^S; \ldots; \mathrm{z}_{N_S}^S] \tag{6}$$

$$Z_N^L = [\mathrm{z}_{\mathrm{cls}}^L; \mathrm{z}_1^L; \ldots; \mathrm{z}_{N_L}^L] \tag{7}$$

where $\mathrm{z}_{\mathrm{cls}}^S$ and $\mathrm{z}_{\mathrm{cls}}^L$ are the classification tokens on the short- and long-term features, respectively, $\mathrm{z}_1^S \ldots \mathrm{z}_{N_S}^S$ are the patch tokens of the short-term features; N_S is the number of short-term patch tokens; $\mathrm{z}_1^L \ldots \mathrm{z}_{N_L}^L$ are the patch tokens of the long-term features; and N_L is the number of long-term patch tokens.

We stack the long- and short-term sub-encoders, M and N, respectively, to construct the deep network and extract deep features.

3.2. Feature cross-attention (FCA)

In the fields of image classification and object detection, a large number of researchers have proposed improved ideas for feature fusion methods (Yu et al., 2020; Zheng et al., 2020), such as feature pyramid networks (FPNs) (Lin et al., 2017), ResNet (He et al., 2016) and adaptive spatial feature fusion (ASFF) (Liu et al., 2019). The above research proved that setting up an appropriate feature fusion strategy is beneficial for improving accuracy. However, the current fusion methods are designed based on the feature maps extracted by convolutional neural networks and are not applicable to the vector features extracted by the transformer model. Therefore, we propose the feature cross-attention module (FCA) to cross-learn the classification token and patch tokens of two branches, which achieves the efficient fusion of long- and short-term features with less computational effort.

Taking the short-term branch as an example, the feature cross-attention module is specified in Figure 4.

First, the short-term feature classification token (CLS token) and the long-term feature patch tokens are aligned and stitched together as in Equations 8, 9:

$$\mathrm{z}_{\mathrm{cls}}^{S'} = \mathrm{f}^S(\mathrm{z}_{\mathrm{cls}}^S) \tag{8}$$

$$\mathrm{z}^{S'} = \mathrm{Concat}(\mathrm{z}_{\mathrm{cls}}^{S'}, \mathrm{z}_1^L, \ldots, \mathrm{z}_{N_L}^L) \tag{9}$$

where $\mathrm{f}^S(\cdot)$ is the feature alignment function and Concat is the splicing operation.

Second, the FCA input $\mathrm{z}_{\mathrm{cls}}^{S'}$ is cross-learned with $\mathrm{z}^{S'}$ as in Equations 10–13.

$$FCA(\mathrm{z}_{\mathrm{cls}}^{S'}, \mathrm{z}^{S'}) = \mathrm{soft\,max}\left(\frac{qk^T}{\sqrt{d}}\right)v \tag{10}$$

where q and k are the query and key of the short-term features, d is the long-term patch token dimension, and Softmax is the normalized exponential function.

Finally, the feature cross-attention is extended to multiple heads, which is denoted as multihead feature cross-attention (MFCA); the multihead features are aligned backward, and their output dimensions are kept consistent with the short-term feature classification token to obtain the short-term feature classification token $Z_{\mathrm{cls}}^{S''}$ after cross-learning, as in Equation 11.

$$\mathrm{z}_{\mathrm{cls}}^{S''} = g^S(\mathrm{z}_{\mathrm{cls}}^{S'} + FCA(\mathrm{z}_{\mathrm{cls}}^{S'}, \mathrm{z}^{S'})) \tag{11}$$

where $\mathrm{g}^S(\cdot)$ is the reverse alignment function and $\mathrm{z}_{\mathrm{cls}}^{S'}$is the classification token before reverse alignment.

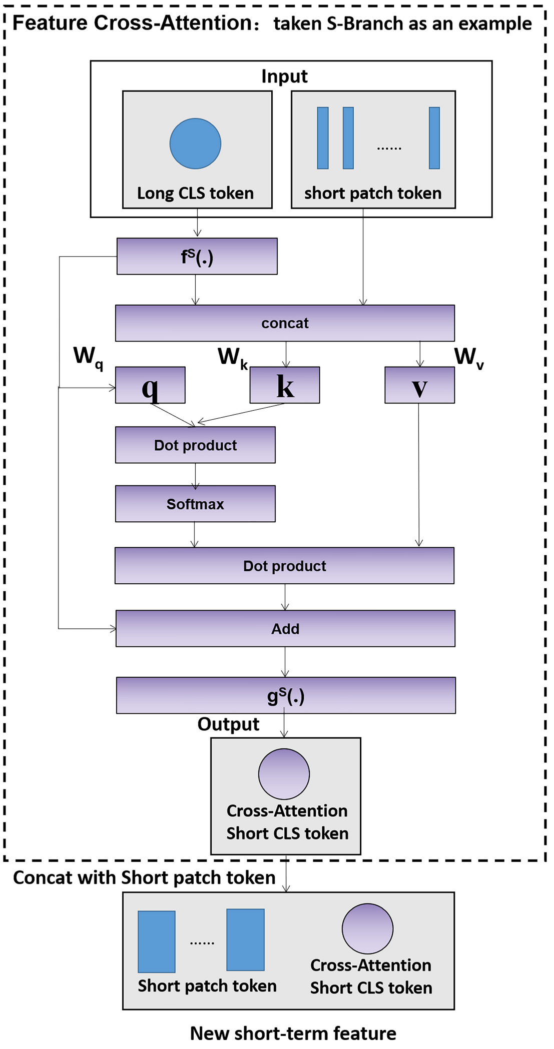

FIGURE 4
Feature cross-attention module.

At this point, the short-term feature after cross-learning is $Z_M^{S'}$, as in Equation 12, and similarly, the long-term feature after cross-learning is $Z_N^{L'}$, as in Equation 13.

$$Z_M^{S'} = [z_{cls}^{S''}; z_1^S; \ldots; z_{N_S}^S] \tag{12}$$

$$Z_N^{L'} = [z_{cls}^{L''}; z_1^L; \ldots; z_{N_L}^L] \tag{13}$$

Since the short-term feature classification token learns the abstract information of the branches, interacting with the patch tokens at the other branches helps to include information at a different scale. The fused long-term and short-term features $Z_M^{S'}$ and $Z_N^{L'}$ are output to the gesture classification model.

3.3. Gesture classification module

The short-term feature $z_{cls}^{S''}$ and long-term feature $z_{cls}^{L''}$ classification tokens are obtained, and the sum of the gesture scores of each branch is output to obtain the gesture category.

$$gestures = LL(LayerNorm(z_{cls}^{S''})) + LL(LayerNorm(z_{cls}^{L''})) \tag{14}$$

4. Experiments and results

Our experiments employed a deep learning framework on a computer platform for model training and testing. The computer hardware configuration used was an Intel Core i7-8700K CPU processor (32 GB RAM) and a GeForce GTX 3090 GPU (24 GB RAM). The operating system was Ubuntu 18.04.4LTS, and network models were constructed, trained, and validated using the Python 3.6.5 programming language under the PyTorch 1.8.0 deep learning framework. The cross-entropy loss was used to measure classification performance.

4.1. LST-EMG-Net model training parameter setting

We evaluated different variants of the LST-EMG-Net architecture. For all model variants, we used the Adam optimizer to set the parameters to 0.9, 0.999, and the learning rate was corrected using StepLR, with the step size set to 3 and gamma set to 0.5. We set the initial learning rate to 6e-4 with a batch size of 512. The short-term patch length and long-term patch length were dynamically set according to the sEMG window length used for the dataset. For the short-term branch, the short-term subencoder depth was set to 1 (i.e., $M = 1$), and the number of short-term subencoder heads was set to eight. For the long-term branch, the long-term subencoder depth was set to 2 (i.e., $N = 2$), and the number of long-term subencoder heads was set to eight. The feature *cross-attention depth* $= 4$ (i.e., $L = 4$), and the number of feature *cross-attention heads* $= 8$. All models were trained under these parameters until convergence.

4.2. Ablation experiments

In this article, we evaluate the LST-EMG-Net model on three datasets and describe the ablation experiments of the LST-EMG-Net model. This ablation experiment used the ViT model extended to dual streams as the baseline and added the gesture recognition effects of FCA, ECA, MHRA, and SA. The baseline model framework was used to remove the yellow module in **Figure 3**. We recorded the average accuracy of all subjects on each dataset to form **Table 1**.

LST-EMG-Net shows the best model results obtained by simultaneously adding FCA, ECA, MHRA, and SA. Model 1 improves the average recognition accuracy by 2.78% compared to the baseline. This demonstrates that dual-stream information fusion helps improve accuracy. Model 2 improved the average recognition accuracy by 2.73% on the three datasets compared to Model 1, with a 4.29% improvement on the CapgMyo DB-c high-density dataset. Because of the high number of channels of the sEMG acquisition device in this dataset, 128 channels, and the rich muscle activity information between channels, the ECA

TABLE 1 Hand gesture recognition accuracy achieved on each dataset in the ablation experiments.

Model name	FCA	ECA	MHRA	SA	DB2 exercise B	DB5 exercise C	CapgMyo DB-c
Baseline					76.25%	77.17%	91.60%
Model 1	√				78.13% (+1.88%)	83.00% (+5.83%)	92.24% (+0.64%)
Model 2	√	√			79.51% (+1.38%)	85.53% (+2.53%)	96.53% (+4.29%)
Model 3	√	√	√		80.62% (+1.11%)	86.96% (+1.43%)	**98.95% (+2.32%)**
Model 4	√	√	√	√	**81.47% (+0.85%)**	**88.24% (+1.28%)**	98.80% (−0.15%)

The bold value means the best recognition accuracy.

effect is evident in this dataset. Model 3 improved the average recognition accuracy by 1.62% compared with model 2. Because the amount of sEMG data in this experiment was smaller and the number of required network layers was relatively shallow, with the addition of MHRA, the network may perform better on more extensive sEMG data. Model 4 achieves an average recognition accuracy improvement of 0.66% compared to model 3. However, on the CapgMyo DB-c high-density dataset, the original signal size reached saturation due to the high sampling rate and the number of channels in this dataset. Therefore, compared to model 3, the model accuracy stabilized.

4.3. Comparison experiment

We compare the proposed LST-EMG-Net with the existing MSCNN of the multistream CNN (Wei et al., 2019), the bidirectional temporal convolutional network (BiTCN) (Chen et al., 2020), and TEMG based on the vision transformer (ViT) (Siddhad et al., 2022) on the above three EMG datasets.

(1) LST-EMG-Net's accuracy and inference time: The performance is shown in Table 2.

From Table 2, we can see that our method reaches the optimum results on the three datasets of DB2 Exercise B, DB5 Exercise C, and CapgMyo DB-C, and the accuracy is improved by 2.70, 4.49, and 0.42%, respectively, compared to the optimum comparison methods.

TABLE 2 Accuracies and inference times of LST-EMG-Net and the comparison algorithms.

Dataset	Model name	Accuracy	Inference time
DB2 exercise B	MSCNN	71.89%	5.60 ms
	BiTCN	65.79%	5.75 ms
	TEMG	78.77%	1.09 ms
	LSTEMGNet [ours]	**81.47%**	6.47 ms
DB5 exercise C	MSCNN	79.14%	7.27 ms
	BiTCN	83.75%	7.29 ms
	TEMG	68.18%	1.18 ms
	LSTEMGNet [ours]	**88.24%**	6.36 ms
CapgMyo DB-C	MSCNN	86.67%	7.78 ms
	BiTCN	98.38%	7.30 ms
	TEMG	92.90%	1.12 ms
	LSTEMGNet [ours]	**98.80%**	6.32 ms

The bold value means the best recognition accuracy.

Regarding the recognition time aspect, we can also see from Table 2 that LST-EMG-Net not only has higher recognition accuracy but also outperforms the CNN-based MSCNN model and the RNN-based BiTCN model in terms of inference time. Both LST-EMG-Net and TEMG are designed based on the transformer model, but the difference is that LST-EMG-Net extends the transformer to a dual-flow structure. Compared with the single-stream structure of TEMG, the average recognition accuracy of LST-EMG-Net is 9.5% higher on the three datasets, which demonstrates improved recognition accuracy and stability. Furthermore, the dual-stream structure of LST-EMG-Net increases the computational and parametric quantities of the model to a certain extent. On average, it is 5.25 ms slower than TEMG, but both can meet the requirements of real-time recognition.

(2) LST-EMG-Net's stability: To verify the stability of LST-EMG-Net in recognizing various types of gestures, we compare the fluctuation of the recognition accuracy of the method in this article with that of MSCNN, BiTCN, and TEMG. We choose the standard deviation (STD) as an indicator to measure the fluctuation of each gesture between subjects. Taking gesture one as an example, the fluctuation value is calculated as follows in Equation 15.

$$G1 = \sqrt{\frac{\sum_{i=1}^{n} acc_i - \overline{acc}}{n}} \tag{15}$$

where i is the subject number, $\mathrm{acc_i}$ denotes the i-th subject gesture one accuracy, $\overline{acc}$ is the average gesture pne accuracy, and n is the number of subjects. A smaller fluctuation value means that the gesture recognition is more stable, and we calculate the average fluctuation value of each gesture in the three datasets, as shown in Table 3.

The experimental results in Table 3 show that the average fluctuation value of the proposed LST-EMG-Net is low for all kinds

TABLE 3 Average fluctuation values of LST-EMG-Net and the comparison algorithms.

Model name	DB2 exercise B	DB5 exercise C	CapgMyo DB-C
MSCNN	0.1795	0.1835	0.1252
BiTCN	0.2232	0.1324	0.0396
TEMG	0.1197	0.1392	0.1045
LST-EMG-Net [ours]	**0.1181**	**0.1098**	**0.0179**

The bold value means the highest recognition stability.

of gestures. It is suitable for recognition tasks because it learns the information of EMG sampling points with different timing lengths, thus maintaining a relatively stable and high recognition rate for gestures of different complexity.

5. Conclusion

Current research gives little attention to the problem of matching the amount of information in features with the amount of information needed to recognize gestures. Here, we propose the LST-EMG-Net-based sEMG gesture recognition method to address the above problems; it is mainly composed of a long short-term encoder and a feature cross-attention module. Our method maintains a high level of accuracy for all types of gesture recognition in both sparse EMG datasets and high-density sEMG datasets. It improves the stability of gesture recognition compared to other network structures.

Our LST-EMG-Net framework can be applied well to recognize various types of gestures by subjects. Nevertheless, due to the individual variability among subjects, LST-EMG-Net is difficult to apply to the intersubject recognition of gestures and has a high burden of use for new subjects, which needs further study in clinical applications. In the future, we will improve the LST-EMG-Net framework to achieve intersubject gesture recognition for controlling exoskeletons or other rehabilitation devices for post-surgical rehabilitation of stroke patients.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: http://ninapro.hevs.ch/node/7.

Ethics statement

Written informed consent was obtained from the individual(s) for the publication of any potentially identifiable images or data included in this article.

Author contributions

WZ and TZ conceived the ideas and designed the methodology, implemented the technical pipeline, conducted the experiments, and analyzed the results. JZ and YW provided the dataset for the experiments. All authors discussed the manuscript, wrote the manuscript, and gave final approval for its publication.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Al-Saegh, A., Dawwd, S. A., and Abdul-Jabbar, J. M. (2021). Deep learning for motor imagery EEG-based classification: A review. *Biomed. Signal Process. Control* 63:102172. doi: 10.1016/j.bspc.2020.102172

Alseed, M. M., and Tasoglu, S. (2022). "Machine learning-enabled classification of forearm sEMG signals to control robotic hands prostheses," in *Proceedings of the 2022 innovations in intelligent systems and applications conference (ASYU)* (Antalya: IEEE). doi: 10.1109/ASYU56188.2022.9925273

Atzori, M., Cognolato, M., and Müller, H. (2016). Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands. *Front. Neurorobot.* 10:9. doi: 10.3389/fnbot.2016.00009

Atzori, M., Gijsberts, A., Castellini, C., Caputo, B., Hager, A. G. M., Elsig, S., et al. (2014a). Electromyography data for non-invasive naturally-controlled robotic hand prostheses. *Sci. Data* 1, 1–13.

Atzori, M., Gijsberts, A., Kuzborskij, I., Elsig, S., Hager, A. G. M., Deriaz, O., et al. (2014b). Characterization of a benchmark database for myoelectric movement classification. *IEEE Trans. Neural Syst. Rehabil. Eng.* 23, 73–83. doi: 10.1109/TNSRE.2014.2328495

Atzori, M., Gijsberts, A., Heynen, S., Hager, A. G. M., Deriaz, O., Van Der Smagt, P., et al. (2012). "Building the Ninapro database: A resource for the biorobotics community," in *Proceedings of the 2012 4th IEEE RAS & EMBS international conference on biomedical robotics and biomechatronics (BioRob)* (Rome: IEEE), 1258–1265. doi: 10.1109/BioRob.2012.6290287

Baygin, M., Barua, P. D., Dogan, S., Tuncer, T., Key, S., Acharya, U. R., et al. (2022). A hand-modeled feature extraction-based learning network to detect grasps using sEMG signal. *Sensors* 22:2007. doi: 10.3390/s22052007

Briouza, S., Gritli, H., Khraief, N., Belghith, S., and Singh, D. (2022). "EMG signal classification for human hand rehabilitation via two machine learning techniques: KNN and SVM," in *Proceedings of the 2022 5th international conference on advanced systems and emergent technologies (IC_ASET)* (Hammamet: IEEE), 412–417. doi: 10.1109/IC_ASET53395.2022.9765856

Chen, H., Zhang, Y., Zhou, D., and Liu, H. (2020). "Improving gesture recognition by bidirectional temporal convolutional netwoks," in *Proceedings of the international conference on robotics and rehabilitation intelligence* (Singapore: Springer), 413–424. doi: 10.1007/978-981-33-4932-2_30

Cipriani, C., Zaccone, F., Micera, S., and Carrozza, M. C. (2008). On the shared control of an EMG-controlled prosthetic hand: Analysis of user–prosthesis interaction. *IEEE Trans. Robot.* 24, 170–184. doi: 10.1109/TRO.2007.910708

Comon, P. (1992). *Independent component analysis. J-L.Lacoume. Higher-order statistics.* Amsterdam: Elsevier, 29–38. doi: 10.1109/TNN.2004.826380

Du, Y., Wenguang, J., Wentao, W., and Geng, W. (2017). *CapgMyo: A high density surface electromyography database for gesture recognition.* Hangzhou: Zhejiang University, 2017.

Farago, E., Macisaac, D., Suk, M., and Chan, A. D. C. (2022). A review of techniques for surface electromyography signal quality analysis. *IEEE Rev. Biomed. Eng.* 16, 472–486. doi: 10.1109/RBME.2022.3164797

Gijsberts, A., Atzori, M., Castellini, C., Müller, H., and Caputo, B. (2014). Measuring movement classification performance with the movement error rate. *IEEE Trans. Neural Syst. Rehabil. Eng.*

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, NV, 770–778. doi: 10.1109/CVPR.2016.90

Hu, Y., and Wang, Q. (2020). "A comprehensive evaluation of hidden Markov model for hand movement recognition with surface electromyography," in *Proceedings of the 2020 2nd international conference on robotics, intelligent control and artificial intelligence*, New York, NY, 85–91. doi: 10.1145/3438872.3439060

Hu, Y., Wong, Y., Wei, W., Du, Y., Kankanhalli, M., and Geng, W. (2018). A novel attention-based hybrid CNN-RNN architecture for sEMG-based gesture recognition. *PLoS One* 13:e0206049. doi: 10.1371/journal.pone.0206049

Huang, Y. Y., Low, K. H., and Lim, H. B. (2009). "Objective and quantitative assessment methodology of hand functions for rehabilitation," in *Proceedings of the 2008 IEEE international conference on robotics and biomimetics* (Bangkok: IEEE), 846–851. doi: 10.1109/ROBIO.2009.4913110

Jia, R., Yang, L., Li, Y., and Xin, Z. (2021). "Gestures recognition of sEMG signal based on Random Forest," in *Proceedings of the 2021 IEEE 16th conference on industrial electronics and applications (ICIEA)* (Chengdu: IEEE), 1673–1678. doi: 10.3389/fnhum.2022.911204

Karim, S., Qadir, A., Farooq, U., Shakir, M., and Laghari, A. (2022). Hyperspectral imaging: A review and trends towards medical imaging. *Curr. Med. Imaging.* doi: 10.2174/1573405618666220519144358

Laghari, A., Estrela, V., and Yin, S. (2022). How to collect and interpret medical pictures captured in highly challenging environments that range from nanoscale to hyperspectral imaging. *Curr. Med. Imaging.* doi: 10.2174/1573405619666221228094228

Leonardis, D., Barsotti, M., Loconsole, C., Solazzi, M., Troncossi, M., Mazzotti, C., et al. (2015). An EMG-controlled robotic hand exoskeleton for bilateral rehabilitation. *IEEE Trans. Haptics* 8, 140–151. doi: 10.1109/TOH.2015.2417570

Li, X., Liu, J., Huang, Y., Wang, D., and Miao, Y. (2022). Human motion pattern recognition and feature extraction: An approach using multi-information fusion. *Micromachines* 13:1205. doi: 10.3390/mi13081205

Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017). "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117–2125. doi: 10.1109/CVPR.2017.106

Liu, L., Chen, X., Lu, Z., Cao, S., Wu, D., and Zhang, X. (2016). Development of an EMG-ACC-based upper limb rehabilitation training system. *IEEE Trans. Neural Syst. Rehabil. Eng.* 25, 244–253. doi: 10.1109/TNSRE.2016.2560906

Liu, S., Huang, D., and Wang, Y. (2019). Learning spatial fusion for single-shot object detection. *arXiv* [Preprint]. doi: 10.48550/arXiv.1911.09516

Menon, R., Di Caterina, G., Lakany, H., Petropoulakis, L., Conway, B. A., and Soraghan, J. J. (2017). Study on interaction between temporal and spatial information in classification of EMG signals for myoelectric prostheses. *IEEE Trans. Neural Syst. Rehabil. Eng.* 25, 1832–1842. doi: 10.1109/TNSRE.2017.2687761

Montazerin, M., Zabihi, S., Rahimian, E., Mohammadi, A., and Naderkhani, F. (2022). ViT-HGR: Vision transformer-based hand gesture recognition from high density surface EMG signals. *arXiv* [Preprint]. doi: 10.48550/arXiv.2201.10060

Mujahid, A., Awan, M. J., Yasin, A., Mohammed, M. A., Damaševičius, R., Maskeliūnas, R., et al. (2021). Real-time hand gesture recognition based on deep learning YOLOv3 model. *Appl. Sci.* 11:4164. doi: 10.3390/app11094164

Muri, F., Carbajal, C., Echenique, A. M., Fernández, H., and López, N. M. (2013). Virtual reality upper limb model controlled by EMG signals. *J. Phys. Conf. Ser.* 477:012041. doi: 10.1088/1742-6596/477/1/012041

Naik, G. R., and Nguyen, H. T. (2014). Nonnegative matrix factorization for the identification of EMG finger movements: Evaluation using matrix analysis. *IEEE J. Biomed. Health Inform.* 19, 478–485. doi: 10.1109/JBHI.2014.2326660

Naik, G. R., Kumar, D. K., Weghorn, H., and Palaniswami, M. (2007). "Subtle hand gesture identification for HCI using temporal decorrelation source separation BSS of surface EMG," in *Proceedings of the 9th biennial conference of the Australian pattern recognition society on digital image computing techniques and applications (DICTA 2007)* (Glenelg, SA: IEEE), 30–37. doi: 10.1109/DICTA.2007.4426772

Namazi, H. (2019a). Decoding of hand gestures by fractal analysis of electromyography (EMG) signal. *Fractals* 27:1950022. doi: 10.1142/S0218348X19500221

Namazi, H. (2019b). Fractal-based classification of electromyography (EMG) signal between fingers and hand's basic movements, functional movements, and force patterns. *Fractals* 27:1950050. doi: 10.1142/S0218348X19500506

Narayan, Y. (2021). Hb vsEMG signal classification with time domain and frequency domain features using LDA and ANN classifier. *Mater. Today Proc.* 37, 3226–3230.

Nazmi, N., Abdul Rahman, M. A., Yamamoto, S. I., Ahmad, S. A., Malarvili, M. B., Mazlan, S. A., et al. (2017). Assessment on stationarity of EMG signals with different windows size during isotonic contractions. *Appl. Sci.* 7:1050. doi: 10.3390/app7101050

Oudah, M., Al-Naji, A., and Chahl, J. (2020). Hand gesture recognition based on computer vision: A review of techniques. *J. Imaging* 6:73. doi: 10.3390/jimaging6080073

Parajuli, N., Sreenivasan, N., Bifulco, P., Cesarelli, M., Savino, S., Niola, V., et al. (2019). Real-time EMG based pattern recognition control for hand prostheses: A review on existing methods, challenges and future implementation. *Sensors* 19:4596. doi: 10.3390/s19204596

Pizzolato, S., Tagliapietra, L., Cognolato, M., Reggiani, M., Müller, H., and Atzori, M. (2017). Comparison of six electromyography acquisition setups on hand movement classification tasks. *PLoS One* 12:e0186132. doi: 10.1371/journal.pone.0186132

Rahimian, E., Zabihi, S., Asif, A., Farina, D., Atashzar, S. F., and Mohammadi, A. (2021). TEMGNet: Deep transformer-based decoding of Upperlimb sEMG for hand gestures recognition. *arXiv* [Preprint]. doi: 10.48550/arXiv.2109.12379

Rim, B., Sung, N. J., Min, S., and Hong, M. (2020). Deep learning in physiological signal data: A survey. *Sensors* 20:969. doi: 10.3390/s20040969

Scheme, E., and Englehart, K. (2011). Electromyogram pattern recognition for control of powered upper-limb prostheses: State of the art and challenges for clinical use. *J. Rehabil. Res. Dev.* 48, 643–659. doi: 10.1682/JRRD.2010.09.0177

Siddhad, G., Gupta, A., Dogra, D. P., and Roy, P. P. (2022). Efficacy of transformer networks for classification of raw EEG data. *arXiv* [Preprint]. doi: 10.48550/arXiv.2202.05170

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. *Adv. Neural Inform. Process. Syst.* 30, 5998–6008.

Wang, B., Yang, C., and Xie, Q. (2012). "Human-machine interfaces based on EMG and Kinect applied to teleoperation of a mobile humanoid robot," in *Proceedings of the 10th world congress on intelligent control and automation* (Beijing: IEEE), 3903–3908. doi: 10.1109/WCICA.2012.6359124

Wei, W., Wong, Y., Du, Y., Hu, Y., Kankanhalli, M., and Geng, W. (2019). A multi-stream convolutional neural network for sEMG-based gesture recognition in muscle-computer interface. *Pattern Recogn. Lett.* 119, 131–138. doi: 10.1016/j.patrec.2017.12.005

Xia, P., Hu, J., and Peng, Y. (2018). EMG-based estimation of limb movement using deep learning with recurrent convolutional neural networks. *Artif. Organs* 42, E67–E77. doi: 10.1111/aor.13004

Xue, Y., Ji, X., Zhou, D., Li, J., and Ju, Z. (2019). SEMG-based human in-hand motion recognition using nonlinear time series analysis and random forest. *IEEE Access* 7, 176448–176457. doi: 10.1109/ACCESS.2019.2957668

Yu, J., Li, H., Yin, S. L., and Karim, S. (2020). Dynamic gesture recognition based on deep learning in human-to-computer interfaces. *J. Appl. Sci. Eng.* 23, 31–38. doi: 10.6180/jase.202003_23(1).0004

Zheng, D., Li, H., and Yin, S. (2020). Action recognition based on the modified twostream CNN. *Int. J. Math. Sci. Comp.* 6, 15–23. doi: 10.5815/ijmsc.2020.06.03

Zhou, D., Kang, B., Jin, X., Yang, L., Lian, X., Jiang, Z., et al. (2021). Deepvit: Towards deeper vision transformer. *arXiv* [Preprint]. doi: 10.48550/arXiv.2103.11886