A BERT Based Method for Continuous Estimation of Cross-Subject Hand Kinematics From Surface Electromyographic Signals

Chuang Lin, Xingjian Chen, *Graduate Student Member, IEEE*, Weiyu Guo, Ning Jiang,
Dario Farina, *Fellow, IEEE*, and Jingyong Su

***Abstract*—Estimation of hand kinematics from surface electromyographic (sEMG) signals provides a non-invasive human-machine interface. This approach is usually subject-specific, so that the training on one individual does not generalise to different subjects. In this paper, we propose a method based on Bidirectional Encoder Representation from Transformers (BERT) structure to predict the movement of hands from the root mean square (RMS) feature of the sEMG signal following μ-law normalization. The method was tested for within-subject and cross-subject conditions. We trained the model with two hard sample mining methods, Gradient Harmonizing Mechanism (GHM) and Online Hard Sample Mining (OHEM). The proposed method was compared with classic approaches, including long short-term memory (LSTM) and Temporal Convolutional Network (TCN) as well as a recent method called Long Exposure Convolutional Memory Network (LE-ConvMN). Correlation coefficient (CC), normalized root mean square error (NRMSE) and time costs were used as performance metrics. Our method (sBERT-OHEM) achieved state-of-the-art performance in cross-subject evaluation as well as high performance in subject-specific tests on the Ninapro dataset. The above tests are based on the same randomly selected 10 subjects. Generally, in the cross-subject situation, with the increasing of the subjects' number, it unavoidably leads to the decline of the performance. While the performance of our method on 38 subjects was significantly higher than the other methods on 10 subjects in cross-subject conditions, which further verified the advantage of our methods.**

***Index Terms*—sEMG, hands kinematics, BERT, hard sample mining, cross-subjects.**

Manuscript received 8 August 2022; revised 2 October 2022; accepted 11 October 2022. Date of publication 21 October 2022; date of current version 30 January 2023. This work was supported by the Guangdong Basic and Applied Basic Research Foundation under Grant 2022A1515010800. *(Chuang Lin and Xingjian Chen contributed equally to this work.) (Corresponding author: Jingyong Su.)*

Chuang Lin is with the School of Information Science and Technology, Dalian Maritime University, Dalian 116026, China.

Xingjian Chen and Jingyong Su are with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen 518055, China (e-mail: sujingyong@hit.edu.cn).

Weiyu Guo is with the Artificial Intelligence Thrust, Hong Kong University of Science and Technology, Hong Kong.

Ning Jiang is with the National Clinical Research Center for Geriatrics, West China Hospital Sichuan University, and the Med-X Center for Manufacturing, Sichuan University, Chengdu 610041, China.

Dario Farina is with the Department of Bioengineering, Imperial College London, SW7 2BX London, U.K.

Digital Object Identifier 10.1109/TNSRE.2022.3216528

I. Introduction

WITH the development of robots, intelligent devices, and the Internet, and the growing need of improving the quality of life of the aging population, interactions between intelligent devices and humans are becoming an important part of our society. As a result, there is a demand for technologies that allow this interaction in a natural, accurate and robust way. For example, human-robot interaction is needed in active prostheses, robot-assisted surgery, drone reconnaissance, and so on. Further, efficient, precise, and user-friendly human-computer interaction (HCI), as well as human-machine collaboration (HMC), has also attracted much attention. To achieve high accuracy and efficiency, the technique of extracting precise features from biological signals and translating them into control commands is playing an important role in HCI and HMC fields. The surface electromyographic (sEMG) signal can be easily recorded with wearable devices and has been used as for decoding human movement intentions for decades [1], [2].

The hand allows humans to perform their most complex movements by a complex structure that provides >20 degrees of freedom [3]. Decoding hand movements by wearable systems would provide a high-information transfer interface.

Recently, deep learning methods have been widely used to select features of sEMG automatically, with excellent performance in classification [4], [5], [6]. Currently, efforts are mainly devoted to continuous movement regression rather than classification.

Current approaches for continuous motion estimation can be roughly divided into two categories, model-based and model-free [7]. Model-based methods are based on physical models including kinematics models [8], musculoskeletal models [9], [10], and dynamic models in general. These models, however, may be very complex and therefore the identification of their parameters may be challenging. Therefore, researchers are currently more inclined to use model-free methods. For instance, a simultaneous and continuous kinematics estimation method was proposed in [11] and used a single ANN for four DoFs across shoulder and elbow joints. In [12], a method was proposed to estimate hand pose from sEMG with recurrent neural networks (RNN) structure. However, to the best of our knowledge, no method can be applied to cross-subject situations.

EMG signals vary from person to person and are even different for the same person at different recording times. Transfer learning methods [13] can be used to adapt a subject-specific model to work on a different subject. For example, Fan et al. [14] proposed a hand gesture recognition method based on transfer learning to learn from models on intact hands to fit amputees. However, transfer learning can add an additional structure that occupies the memory [15] and the output model can only transfer from one subject to another, while it is still difficult to generate a model to be applied on two or more subjects concurrently.

As another challenge, normalization is widely used in deep learning methods. Min-max normalization is the most widely used normalization method in the field of estimation from sEMG. However, sEMG signals of different subjects lie in different ranges with different distributions, in which min-max normalization can disturb their own features when multiple subjects are analysed together. In addition, a large amount of useful information of sEMG in the time domain lie near zero [16] and cannot be identified by linear method.

Finally, the existing regression approaches always ignore studying hard samples containing useful information, which can be continuously concentrated in several time periods [17].

Here, we propose a method based on the Bidirectional Encoder Representation from Transformers (BERT) structure with μ-law normalization, a nonlinear normalization, which can better magnify the low magnitude and keep the scale of larger values. Gradient Harmonizing Mechanism (GHM) and Online Hard Samples Mining (OHEM) are applied to make the models better learn from hard samples, and a smooth layer is applied to reduce the fluctuations caused by BERT. Our methods were validated on the Ninapro dataset and compared with two classical methods, LSTM [18] and TCN [19], as well as a recent novel method called Long Exposure Convolutional Memory Network (LE-ConvMN) [17], which should reach state-of-the-art performance in continuous hand kinematic estimation. In summary, the main contributions of this paper are:

- The BERT-based method was proposed for continuous hand movement regression for the first time.
- A strategy of hard sample mining was applied for better and stable estimation from sEMG.
- Our method can be applied in cross-subject situations, which is not solved by previous works to the best of our knowledge, and the experimental results show that our method reaches state-of-the-art performance.

II. Related Work

In this section, two classical models including LSTM and TCN as well as a recent method named LE-ConvMN for continuous motion estimation will be introduced. Only model-free methods are considered as they are more commonly utilized in practice.

A. Long Short-Term Memory (LSTM)

LSTM [18] is developed from the RNN structure, which has a high capability to solve temporal series problems. RNN allows retaining contextual information gathered at previous iterations to benefit future iterations. However, as the complexity of data and sequence length increases, the short recurrent circles perform weak on long-time series processing. LSTM was designed to solve this problem by a combination of remembering and forgetting. LSTM can naturally remember due to the basic RNN structure and it achieves the forgetting ability by applying the forget gate structure. LSTM is widely used in continuous motion estimation as a classical model-free method because of its long-time memory feature.

B. Temporal Convolutional Network (TCN)

TCN [19] is a neural network that is widely used to extract features of temporal information without RNN structure. The classical convolution network is not suitable for temporal series handling problems due to the limitation of the kernel size.

TCN is established on two basic principles:(1) The network produces an output of the same length as the input. (2) There can be no leakage from the future into the past. For principle (1), the TCN designs a 1D fully-convolutional network (FCN) [20] architecture, whose length of each hidden layer is the same as that of the input layer. For principle (2), the TCN utilizes causal convolution, which only uses the information before the time point the network is predicting.

TCN applies the dilated convolutions [21] to enlarge the receptive field to deal with the long-history temporal series, thus we can avoid stacking too many CNN layers. TCN is another widely used model in continuous motion estimation as a classical model-free method.

C. LE-ConvMN

Long Exposure Convolutional Memory Network (LE-ConvMN) is proposed by Guo et al. [17] to better utilize spatiotemporal information in sEMG data, which is a novel method for continuous hands motion estimation.

Long exposure is a sEMG data processing method. Traditionally, the RMS feature is extracted by a sliding window stepping in window size. The long exposure method extracts features by decreasing the step size, which can be a unique method to increase the quantity of data. The stepping size in this paper is fixed at 1.

ConvLSTM model [22] is then applied to the long exposure data. ConvLSTM is an RNN structure model derived from LSTM. To tackle high-dimensional data, the model develops the fully connected matrix operation of the gates into a convolution operation. LE-ConvMN is claimed to reach state-of-the-art result in this field in [17].

III. Smoothed BERT With Hard Sample Mining

A. BERT

Although all of the models mentioned above are feasible in our field, there are still some flaws. As for the previous deep

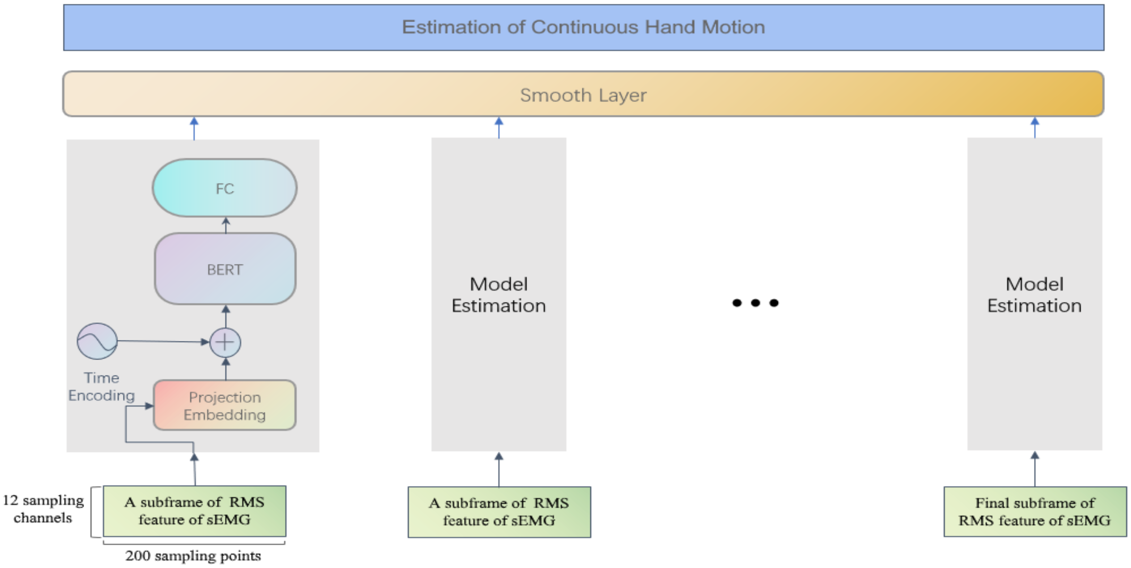

Estimation of Continuous Hand Motion
Smooth Layer
FC
BERT
Time Encoding
Projection Embedding
Model Estimation
Model Estimation
12 sampling channels
A subframe of RMS feature of sEMG
A subframe of RMS feature of sEMG
Final subframe of RMS feature of sEMG
200 sampling points

Fig. 1. Model structure of BERT-based method for continuous hand kinematic estimation from sEMG signals. The raw sEMG signals are extracted to the long-exposure RMS features, which can be seen as a method to augment the data. Consequently, the features are divided into slices with 100ms (200 sampling points) length. Model estimates from every slice of RMS feature after embedding. Finally, all estimated results are smoothed by the smooth layer.

learning methods, LSTM, due to its RNN structure, which makes the training rely on the previous data, so we must keep the order of the data, which leads to time-consuming. TCN does not rely on RNN structure, but on CNN structure, which can lead to instability and fluctuation in the estimation. Although LE-ConvMN can make better use of spatiotemporal information, it is more time-consuming and requires too much memory from the GPU. The cost of training such a model is huge and the hardware requirements are relatively high. Additionally, these methods are all derived from unidirectional structure, which can omit the context information of sEMG.

Bidirectional Encoder Representation from Transformers (BERT) neural network [23] is a novel method in recent years based on transformer [24] structure. BERT has attracted much attention since it was once proposed. Different from RNN and TCN structures, BERT extracts and learns features from series bidirectionally and makes BERT extract features from small-scale temporal and spatial series information, which makes it outperform other models.

BERT is also feasible on multiple subjects due to its strong capability of extracting features from small-scale sequence data brought by attention mechanism and residual skip connections. BERT can recept the future signals and previous signals at the same time, so it can extract features from the whole sequence but not the previous signals only, which makes BERT excelled classical TCN and RNN models on multiple subjects. The performance of BERT on sEMG is shown in the experiments section.

In addition, BERT is an excellent pre-trained method. A well-trained BERT model is claimed to achieve state-of-the-art performance in downstream tasks after fine-tuning in [23], which can contribute to transfer learning in the field.

Transformer is a novel language model to solve the language sequence. The strong ability to extract features of the transformer depends on its attention mechanism. BERT can be viewed as a complicated stack of transformer encoders. Several modifications are designed to BERT to make it more suitable for the estimation problem of hand kinematic series from sEMG. The whole procedure is shown in Figure 1 and described as follows.

1) Model Structure: A transformer encoder consists of two parts, a Multihead Self-Attention Mechanism (MSA) and a multilayer perceptron (MLP) module. To better describe the structure, we denote the 1-D input as $X = [x_0, x_1, \cdots, x_t]$, the output series of the i-th encoder layer as $Z_i (i = 1, \cdots, L)$, L is the numbers of encoder layers. Before feeding input into these encoder layers, we perform embedding on it and designate it as Z_0:

$$Z_0 = [x_0^p, x_1^p, \cdots, x_t^p, \cdots] + E_{time} \tag{1}$$

where x_i^p represents the results of input X after a projection embedding, E_{time} is the time embedding vector, which makes the model has the ability to capture the temporal information of sEMG. A trainable one-dimensional vector is utilized as the time embedding, and we use a linear layer (LL) as linear projection. Thus,

$$X^p = [x_0^p, x_1^p, \cdots, x_t^p, \cdots] = LL_{(s\times c_i, s\times d_h)}(X) \tag{2}$$

where d_h is the embedding size, and naturally, $E_{time} \in \mathcal{R}^{s\times h}$.

Layer normalization as well as residual skip connections are applied in encoder module, to address the degradation problem, thus, an encoder layer can be described as follows:

$$\begin{aligned} Z_l' &= MSA(LayerNorm(Z_{l-1})) + Z_{l-1} \\ Z_l &= MLP(LayerNorm(Z_l')) + Z_l' \end{aligned} \tag{3}$$

where $l \in \{1, 2, \cdots, L\}$, finally, we apply a LL to extract results from Z_L:

$$Z = LL_{(s\times h, 1\times c_o)}(Z_L) \tag{4}$$

Then we introduce the MSA module and the MLP module as follows.

SA can be seen as a process that finds the relation between different sampling points in input Z_i, which is achieved by three matrices named respectively Queries matrix denoted as Q; key matrix denoted as K; values matrix denoted as V. They are calculated by linear transformation:

$$[Q_i, K_i, V_i] = Z_{i-1} W_i^{QKV} \tag{5}$$

where the subscript i means the parameters are computed in encoder layer i, W_i^{QKV} is a learnable weight matrix in each layer. Q, K will be scaled and then compute the weight of V and finally take the weighted sum of all values of V to get the result:

$$SA(Z_{i-1}) = \mathrm{softmax}(\frac{Q_i K_i^T}{\sqrt{d_h}})V \tag{6}$$

where $SA(Z_{i-1}) \in \mathcal{R}^{s \times d_h}$. This method makes the model focus on the important parts of a sEMG input series.

We apply h attention heads to compute Q, K, V in MSA, which means that MSA allows the model to attend to parts of the input sEMG series differently with the different attention heads. MSA concatenates all the outputs of SA computed by different attention heads and then projects it to the result. The MSA can be depicted as follows:

$$\begin{aligned} &MSA(Z_{i-1}) \\ &\quad = [SA_1(Z_{i-1}), SA_2(Z_{i-1}), \cdots, SA_h(Z_{i-1})] W_i^{MSA} \end{aligned} \tag{7}$$

where each SA has its unique Q, K, V, W_i^{MSA} is a learnable weight matrix in each encoder layer.

Moreover, the MLP module can be expressed as follows:

$$\begin{aligned} Z_1' &= LL_{(s \times d_h, s \times d_{fh})}(Z') \\ Z_2' &= \mathrm{GELU}(Z_2') \\ Z_o' &= LL_{(s \times d_{fh}, s \times d_h)} \end{aligned} \tag{8}$$

where d_{fh} is the hidden size of MLP module, and GELU is the Gaussian Error Linear Unit activation function.

2) Smoothing Layer:

BERT is proposed for dealing with language sequence problems, which produces discrete tensors as output. However, joint angles in movements should be successive values. As a result, BERT can cause severe fluctuations in predicted values, although it performs well in the two measure criteria. Therefore, we apply a smoothing method after BERT prediction to get more smooth results to solve this problem.

The results are smoothed by applying a sliding window. For each sliding step, we calculate the average value of all the sampling points in the window:

$$\mathrm{AvgSmooth}(Y) = [\sum_{i=1}^{w} y_i/w, \sum_{i=2}^{w+1} y_i/w, \cdots, \sum_{i=k}^{w+k-1} y_i/w, \cdots] \tag{9}$$

where w stands for the size of sliding window, and $Y = [y_0, y_1, \cdots, y_k, \cdots]$ is the input series.

B. μ-Law Normalization

The μ-law normalization [16], [25] are applied to the RMS feature of sEMG before feeding into models, which is proved that improved performance could be achieved using normalization of the sEMG signals with the μ-law approach [6], [16], [26]. The μ-law normalization is given as:

$$F(x_t) = \mathrm{sign}(x_t) \frac{\ln(1 + \mu |x_t|)}{\ln(1 + \mu)} \tag{10}$$

where x_t means the input at the t-th sampling point, the hyperparameter μ decides the range after normalization.

sEMG signals have the characteristic that many useful information lies near zero, μ-law normalization can magnify the outputs of sensors with small magnitude in a logarithmic fashion, which nonlinear normalization could perform better than linear normalization. The improvement of regression by μ-law normalization is shown in the experiments section.

C. Hard Sample Mining

Hard sample mining is an important problem in data mining. Hard samples are those whose loss is moderately large between estimated values and true values, which can contribute more to model training than easy samples. Comparatively, there is little deviation between the estimated value and the true value of easy samples. In addition, there will inevitably be some bad data, called outliers. The errors that occur when collecting data can also affect the results. Equipment error inevitably leads to outliers in sEMG collecting, and hard samples always occur in datasets. GHM makes models benefit more from these hard samples, but as little as possible from simple samples and outliers, strengthening the robustness and stability of the model.

There are several popular methods for handling hard samples. The simplest way is to increase the size of dataset, but continuous motion data of people is difficult to collect in our field, even though the NinaPro database has a limited amount of data. There are also some low-priced methods, such as clipping, flipping and rotating. Due to the distinctive personal characteristics, these methods cannot effectively augment the sEMG signal. Online Hard example mining (OHEM) [27] is another related method. Hard examples are selected by sorting the input features by loss and taking the several examples for which the current network performs worst in [27] with OHEM. The network is only updated by the selected hard samples.

1) OHEM-MSE Loss:

Hard examples are selected by sorting the output sampling points by MSE loss and taking the several examples for which the current network performs worst with OHEM in this paper. We only update our models with the selected hard samples.

2) GHM-MSE Loss:

Taking outliers into account, loss computing tactics that focus on hard sample mining are applied. Gradient Harmonizing Mechanism (GHM) is a loss calculating mechanism [28], which can assign larger weight to hard samples to make models benefit as much as possible from them. GHM defines gradient norm to measure the deviation between true values and estimated values. We denote the gradient norm as g. g ranges from 0 to 1, when g approximates to 0, the estimated value is almost the same as the true value, while g approximates to 1 means that the two values have a huge deviation.

GHM defines gradient density (GD) to describe the distribution of data by gradient norm g. The relationship can be expressed as follows:

$$GD(g) = \frac{1}{l_\epsilon(g)} \sum_{k=1}^{N} \delta_\epsilon(g_k, g) \tag{11}$$

where GD of g means that the number of sampling points lying in the region centered at g with a length ϵ. We denote the gradient norm of the k-th sampling points in a subframe as g_k. $l_\epsilon(g)$ means the length of actual region, which can be defined as follows:

$$l_\epsilon(g) = \min(g + \epsilon, 1) - \max(g - \epsilon, 0) \tag{12}$$

$\delta_\epsilon(g_k, g)$ is a judging function used for counting, which counts all the sampling points whose gradient norm within the range $l_\epsilon(g)$. The formula is as follows:

$$\delta_\epsilon(g_k, g) = \begin{cases} 1 & \text{if } g_k - \frac{\epsilon}{2} \le g \le g_k + \frac{\epsilon}{2} \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

Then the gradient density harmonizing parameter is defined as:

$$\beta_i = \frac{N}{GD(g_i)} \tag{14}$$

where N is the total number of sampling points in a subframe, β_i is the gradient density harmonizing parameter of the i-th sampling point in the subframe.

We use MSE loss as the base loss in this study, which is defined as follows:

$$L_{MSE} = \frac{\sum_{i=1}^{N} (\theta_i - \hat{\theta}_i)^2}{N} \tag{15}$$

g_i is defined as follows in GHM-MSE loss to make loss distribute sparse respectively because of the lack of data in this field:

$$g_i = \sigma(\frac{2(\theta_i - \hat{\theta}_i)}{N}) \tag{16}$$

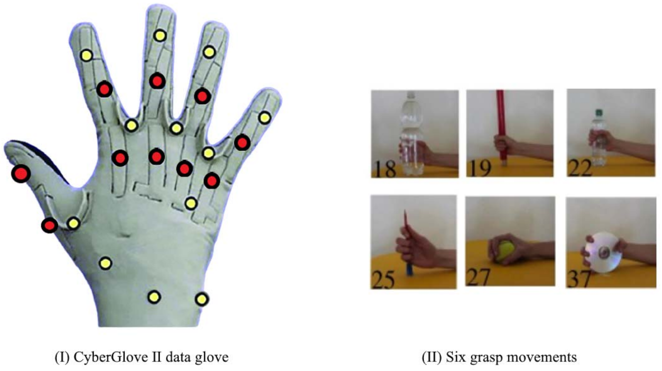
where σ is the sigmoid activation function. Then we define $L_{GHM-MSE}$ as:

$$L_{GHM-MSE} = \frac{\sum_{i=1}^{N} \beta_i (\theta_i - \hat{\theta}_i)^2}{N} \tag{17}$$

IV. Experiments

A. Dataset

Ninapro [29] is a widely used dataset that represents the largest data collection effort with hands intact or amputated in the sEMG field. In the Ninapro dataset, the raw signal of sEMG is sampled with the Delsys Trigno Wireless System, which contains 12 electrodes. Hand kinematics are measured by 22 joint angles and sampled with CyberGlove II data gloves. The sampling rate of sEMG is 2 kHz. The hand kinematics are sampled at a rate of 20 Hz and then resampled to 2 kHz.



(I) CyberGlove II data glove

(II) Six grasp movements

Fig. 2. CyberGlove channels are shown on the left of the figure, where the red dots represent the ten joints to be estimated. Six hand movements were selected and shown on the right of the figure.

1) Subjects Selection: Ninapro includes over 300 data acquisitions divided into 10 datasets that provide electromyography, kinematics and so on. Experiments proceed with selected 10 representative subjects from Ninapro DB2, which includes six repetitions of 49 different movements performed by 40 intact subjects. Our selection makes sure that gender and laterality distribution are relatively uniform to ensure that the method is universal efficacious on different subjects, and their height ranges from 150-187cm, and weight ranges from 52-87kg. Six movements for grasping different objects are chosen for each subject, which have relatively better data quality. Moreover, we concentrate our estimation on 10 finger joints. Our selection is shown in Figure 2.

2) Data Preprocess: We compute Root Mean Square (RMS) with a sliding window of 100 ms at the step of 0.5 ms as the feature. RMS can decrease the noise caused by collecting data. The feature are normalized by μ-law normalization with $\mu = 2^{20}$ after a briefly hyperparameter search experiments for a better performance.

After processing the sEMG, $X \in \mathcal{R}^{s \times c_i}, X = [x_0, x_1, \cdots, x_s]$ denotes the result, where s is the number of sampling points of an input subframe and c_i stands for the number of sEMG channels. Here, $s = 200$, $c_i = 12$. Similarly, $Z \in \mathcal{R}^{1 \times c_o}$ denotes the final output, where c_o is the number of hand kinematic channels. Ten joints of fingers are selected as 10 typical kinematic channels, so $c_o = 10$.

In all subject-specific and cross-subject cases, 7/10 of each subject was used for training and 3/10 for testing. Specifically, each subject was trained and tested individually in subject-specific cases. While in cross-subject cases, we trained the model based on the training data from 10 individuals simultaneously but evaluated the test data from each subject individually for an average performance.

B. Evaluation of Parameters

To evaluate our method and compare it with other methods, two criteria are introduced as follows.

1) Pearson Correlation Coefficient: Pearson correlation coefficient (CC) is a commonly used standard to measure how two variables relate to each other linearly. The value of CC ranges from -1 to 1. The larger the CC value, the more similar

TABLE I

AVERAGE PERFORMANCE OF DIFFERENT MODELS ON 10 JOINTS AND 6 MOVEMENTS OF 10 DIFFERENT SUBJECTS WITH MIN-MAX NORMALIZATION

Model	CC	NRMSE	κ	Time Cost/epoch(s)
LE-LSTM	0.8428	0.0835	1.1456	25.779
LE-TCN	0.8021	0.0934	1.4033	15.469
LE-ConvMN	**0.8872**	**0.0705**	0.6345	13.446
BERT	0.8523	0.0813	1.5133	**4.915**
sBERT	0.8522	0.0811	0.5385	4.918
BERT-GHM	0.8532	0.0807	1.5480	4.922
BERT-OHEM	0.8531	0.0806	1.5282	4.924
sBERT-GHM	0.8533	0.0804	**0.5382**	4.929
sBERT-OHEM	0.8536	0.0802	0.5438	4.931
LE-LSTM*	0.4950	0.1661	–	255.057
LE-TCN*	0.5398	**0.1467**	–	152.045
LE-ConvMN*	**0.5797**	0.1517	–	130.467
BERT*	0.4504	0.1927	–	44.019
sBERT*	0.4574	0.1866	–	**44.017**
BERT-GHM*	0.4195	0.1966	–	45.646
BERT-OHEM*	0.5275	0.1899	–	44.111
sBERT-GHM*	0.4297	0.1912	–	45.933
sBERT-OHEM*	0.4539	0.1751	–	44.192

1. '*' represents the model is trained on all the selected subjects.

TABLE II

AVERAGE PERFORMANCE ON DIFFERENT MODELS ON 10 JOINTS AND 6 MOVEMENTS OF 10 DIFFERENT SUBJECTS ON 10 JOINTS AND 6 MOVEMENTS OF 10 DIFFERENT SUBJECTSWITH μ-LAW NORMALIZATION

Model	CC	NRMSE	κ	Time Cost/epoch(s)
LE-LSTM	0.7685	0.0960	0.5812	26.360
LE-TCN	0.8329	0.0875	1.5332	15.623
LE-ConvMN	**0.9017**	**0.0648**	0.6883	13.468
BERT	0.8678	0.0773	1.5708	**4.948**
sBERT	0.8675	0.0772	0.5348	4.951
BERT-GHM	0.8679	0.0770	1.5272	4.956
BERT-OHEM	0.8683	0.0768	1.5857	4.957
sBERT-GHM	0.8689	0.0766	0.5435	4.962
sBERT-OHEM	0.8693	0.0764	**0.5323**	4.960
LE-LSTM*	0.5536	0.1495	–	252.660
LE-TCN*	0.6695	0.1292	–	151.598
LE-ConvMN*	0.7410	0.1229	–	130.060
BERT*	0.8346	0.0881	1.6098	**44.001**
sBERT*	0.8427	0.0850	**0.6011**	44.426
BERT-GHM*	0.8308	0.0877	1.5815	46.386
BERT-OHEM*	0.8299	0.0878	1.6019	44.039
sBERT-GHM*	**0.8441**	0.0846	0.6012	45.569
sBERT-OHEM*	0.8430	**0.0835**	0.6263	44.449
sBERT-OHEM**	0.8065	0.0911	–	165.454

1. '*' represents the model is trained on all the 10 selected subjects.
2. Further, sBERT-OHEM, as our method, was verified on all the 38 subjects in Ninapro DB2 dataset in cross-subject situation, except for two possibly corrupted subject data, denoted with a superscript '**'.

the predicted movement is to the estimated movement, which means that we get a better estimation.

2) Normalized Root Mean Square Error: Root Mean Square Error (RMSE) is a typical measure of the deviation between predicted values and the values actually observed. For the same joint angle, the smaller the RMSE, the better our estimation is. However, we cannot compare the RMSE of different joint angles. Min-max normalization on RMSE is used to solve this problem. Hence, the Normalized RMSE (NRMSE) is defined as:

$$NRMSE = \frac{RMSE}{\theta_{max} - \theta_{min}} \tag{18}$$

where $\theta_{max}, \theta_{min}$ represent the maximum and minimum true value of angles of a certain joint.

3) Unbiased Standard Deviation: Unbiased Standard Deviation (denoted as σ) is a frequently used criterion to measure the dispersion degree of a group of data. The σ of 10 joints of each subject is produced to measure the stability of each model. The value is smaller, the dispersion degree is lower, and the stability of estimation is better. This criterion is adopted in our work based on the results of 10 predicted joints for each subject.

4) Average Curvature: The mean curvature (denoted as κ) of all points of each joint is adopted to measure the smoothness of an estimated curve. The smaller the curvature is, the smoother the curve is.

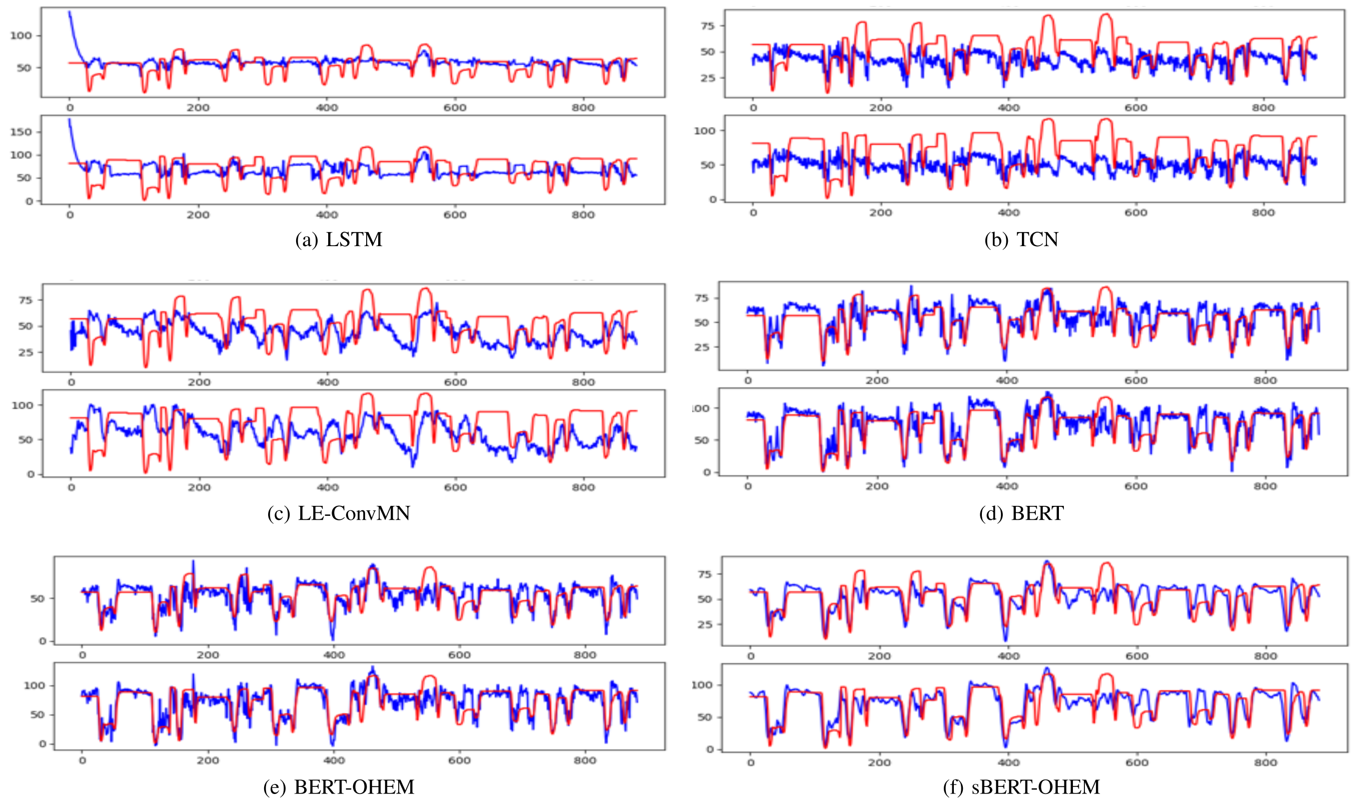
C. Experimental Results

The efficiency and accuracy of our method were validated and compared with previous models on continuous hand movement estimation tasks. All the models were applied on Pytorch framework [30].

All the models were trained on the same GPU (NVIDIA GeForce RTX 3090), and every model was trained for 400 epochs except LE-ConvMN trained for 1000 epochs for more epochs to convergence. Long exposure method [17] was utilized for every model to exclude the influence of data processing. Experiments were carried out in two types of situations: subject-specific and cross-subject situations. The models were trained and verified on data from a single subject in subject-specific situations. In contrast, models were simultaneously trained on data from multiple subjects and verified on every single subject. The NRMSE and CC of each model were calculated on each subject to show the performance of the model in detail. Likewise, we counted the average training time per epoch and estimated the convergence time in each training operation. Inference time is the time cost of estimation in practice, the process was simulated on the same CPU (Intel i7-10875H) to compare the performance of different models. For these criteria, the Friedman test and Wilcoxon signed-rank test were applied to evaluate the significance of our method, and the results were corrected by Bonferroni correction.

After trying several groups of parameters, we selected a relatively better group of parameters to proceed with our experiments. All models were trained at a learning rate of 0.0001 and cut in half after 200 epochs or after every 300 epochs for LE-ConvMN only.

To validate the superiority of BERT, we conducted training on both RNN and TCN models for 10 subjects and 10 channels. And the results are as Table I shows. Prefix 's-' of the model name means that a smooth layer was applied, and the suffix '-GHM' or '-OHEM' after the model name means that this model trained with GHM or OHEM in current and following experiments. Prefix 'LE-' is used to distinguish our models from bare classical models.



(a) LSTM

(b) TCN

(c) LE-ConvMN

(d) BERT

(e) BERT-OHEM

(f) sBERT-OHEM

Fig. 3. Estimation results of BERT-based method and other methods in cross-subject situations. Our method was trained with μ-law normalization while others were trained with classical Min-Max normalization. Red curves are the ground truth and blue curves are the estimation. Only two joints are shown in the figure to better show the detail of the estimation, which can represent the quality of the estimation of all 10 joints.

Among the 10 subjects, subject S3 had the best performance. We show the results of two representative joint angles of S3 in Figure 3, as representations of our method, BERT, BERT-OHEM, and sBERT-OHEM with μ-law normalization are shown in Figure 3.

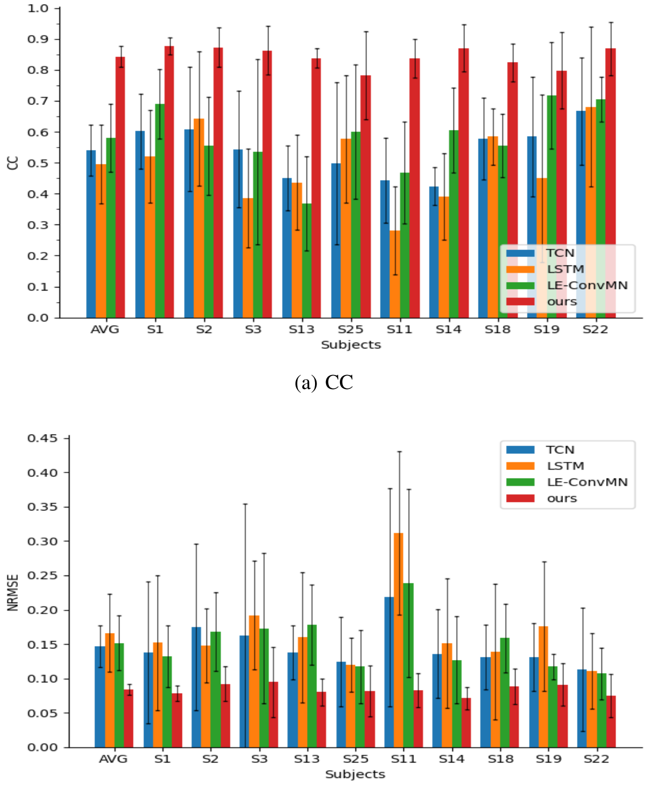
The average CC and NRMSE of our method (The best performance 0.87 ± 0.05; 0.07 ± 0.01) of single subject was significantly better than LE-TCN (0.80 ± 0.06, $p = 0.005$; 0.09 ± 0.02, $p = 0.005$), mildly better than LE-LSTM (0.84 ± 0.11, $p = 0.262$; 0.08 ± 0.03, $p = 0.241$), but slightly worse than LE-ConvMN (0.89 ± 0.12, $p = 0.074$; 0.07 ± 0.03, $p = 0.059$). The average training time cost of our method is significantly lower than that of others due to its special structure, which makes it impossible to train in parallel. BERT needs less average time to reach convergence (about 0.5h) than LE-LSTM (about 2.5h), LE-TCN (about 1.5h), LE-ConvMN (about 1.8h). In addition, the strong capability of extracting features of BERT-based method from sEMG series was verified.

To compare the μ-law Normalization and Min-Max Normalization, they were applied to the every model for each subject we chose, respectively. μ was set as 2^{20} in our study. The result is in Table I and Table II, which shows that μ-law Normalization performs better than Min-Max Normalization in our study, leading to significant improvement.

Based on the BERT training with μ-law Normalization, we applied both GHM or OHEM and smooth layer on models, thus designing some variants, whose results are shown in Table II. Besides, LE-ConvMN, LE-LSTM and LE-TCN with μ-law normalization were validated in the cross-subjects experiments to exclude the influence caused by normalization.

As the result shows, both GHM, OHEM, and smooth layer led to a slight improvement in both single and multiple subjects. Although there were few improvements in criteria, OHEM and smooth layer improved the stability of estimation and reduced the fluctuation of the predicted motion curve, which is shown in Figure 3. OHEM with a smooth layer performed the best in the variants of BERT-based models. GHM led to descending on stability. LE-ConvMN still has state-of-the-art performance on a single subject, but our method can reach better stability in estimation, which is shown in the following paragraph. Figures of two criteria are shown in Figure 4 to show the performance of estimation on every single subject. Additionally, when increasing the number of subjects in cross-subject situations, there was an acceptable decline in the criteria. Although BERT-based methods sufferred more from fluctuation, the smooth layer could effectively preclude it as shown in the Table I, Table II and Figure 3. The performance of our method on even 38 subjects was better than that of other methods on only 10 subjects, which shows the strong capability of extracting features in cross-subject situations.

As shown in Figure 4, unbiased standard deviation of estimation on 10 joints was introduced to evaluate the stability of estimation of these methods and was denoted as σ (σ_c, σ_n for CC and NRMSE, respectively). Our method in subject-specific



(a) CC

(b) NRMSE

Fig. 4. Summary of performance criteria of 10 healthy subjects with intact hands in cross-subject situations. The results show that our method outperforms other model-free methods in all subjects. CC and NRMSE of our method are significantly higher than other models. In addition, one can see from the deviation bar that our method is more stable than others.

situations, sBERT-OHEM ($\sigma_c = 0.0614$; $\sigma_n = 0.0242$;) as the represent, had the significant better performance than TCN ($\sigma_c = 0.0856$, $p = 0.047$; $\sigma_n = 0.0268$, $p = 0.766$) in σ_c, mildly better in σ_n; mildly lower than LSTM ($\sigma_c = 0.0718$, $p = 0.384$; $\sigma_n = 0.0254$, $p = 0.859$) in both two criteria and mildly lower than LE-ConvMN ($\sigma_c = 0.0802$, $p = 0.683$; $\sigma_n = 0.0301$, $p = 0.574$) in σ_c and σ_n. Our method in cross-subjects situations, sBERT-OHEM ($\sigma_c = 0.0753$; $\sigma_n = 0.0265$) as the represent, had the significant lower σ in both two criteria than TCN ($\sigma_c = 0.1570$, $p = 0.007$; $\sigma_n = 0.0929$, $p = 0.005$), LSTM ($\sigma_c = 0.1785$, $p = 0.005$; $\sigma_n = 0.0826$, $p = 0.005$) and LE-ConvMN ($\sigma_c = 0.1588$, $p = 0.007$; $\sigma_n = 0.0628$, $p = 0.009$). And the average unbiased standard deviations of each model in both situations are shown in Table III. The results indicate that our method has better stability than other models.

Since different subjects have different features, EMG signals have subject-specific and non-stationary characteristics, which has always been difficult for previous methods to discover a universal method to fit cross-subject situations. Thus, researchers always train and customize a unique set of model parameters for a certain subject. However, since the strong capability of extracting features from small-scale data of BERT, it can perform better than other methods on multiple subjects. In the following experiment, sEMG signals of all ten selected subjects were concatenated together. We compared our method with LE-LSTM, LE-TCN and LE-ConvMN methods on the combination of subjects to validate the performance of different models on multiple subjects. The result is shown in Table I and Table II.

TABLE III
UNBIASED STANDARD DEVIATION AMONG 10 JOINTS OF DIFFERENT MODELS

Model	σ_c	σ_n	σ_c*	σ_n*
LE-LSTM	0.0718	0.0254	0.1785	0.0826
LE-TCN	0.0856	0.0268	0.1570	0.0929
LE-ConvMN	0.0802	0.0301	0.1588	0.0628
sBERT-OHEM	**0.0614**	**0.0242**	**0.0753**	**0.0265**

1. BERT-based models utilized the μ-law normalization while min-max normalization was applied on TCN, LSTM and LE-ConvMN.
2. '*' represents the results are verified in cross-subjects situations.

TABLE IV
INFERENCE TIME (IT) OF DIFFERENT MODELS ON THE INTEL I7-10875H CPU

Model	TCN	LSTM	LE-ConvMN	BERT-based	sBERT-based
IT(ms)	**3.99**	48.54	18.23	31.78	32.22

1. OHEM and GHM don't affect the inference while smooth layer leads to additional consumption of time. We divided the variants of BERT into two parts as BERT-based and sBERT-based in the table by whether applying smooth layer.

The results in Table I and II indicated that our method significantly outperformed the other models in cross-subject situations. When Min-Max normalization was applied, BERT-based models and LE-ConvMN performed equally. However, while μ-law normalization brought significant improvement to BERT-based models, it had nearly no effect on the performance of LE-ConvMN. As a result, our method significantly outperformed all other models and achieved state-of-the-art performance in cross-subject situations. GHM and OHEM performed near equally on multiple subjects with mild improvement, while they led to an unstable effect on the performance of different single subjects. The more sampling points there are, the higher the possibility that GHM works better. Compared to subject-specific models, cross-subject models inevitably lead to a decline in performance, but it is acceptable. However, the cross-subject model still does not work well on more subjects except those for training.

Inference time is the time for the model to estimate the motion from sEMG. We perform all methods on the test dataset of subject S1 on the same CPU (Intel i7-10875H). Average values were adopted to evaluate performance. The results are shown in Table IV.

TCN is the most efficient method due to the results, while BERT-based method only outperformed LSTM because inference operations cannot be performed in parallel, which made BERT lose its superiority in the time cost. LE-ConvMN has fewer parameters than LSTM, which makes it faster than LSTM and BERT-based method.

In conclusion, BERT with smooth layer and trained with OHEM mechanism has the best performance among

BERT-based variants. Our method not only outperforms the TCN in quality but also stability in both subject-specific situations and cross-subject situations. Although TCN infers faster, it is absent in quality. Additionally, our method outperforms LSTM in all the criteria, including quality, efficiency and stability in both quality and efficiency in both subject-specific situations and cross-subject situations. Our method performs equally to LE-ConvMN in quality in subject-specific situations, while having lower efficiency in inference. However, BERT-based method can train in parallel, which allows us to get a subject-specific model faster and have higher estimation stability. Our method and LE-ConvMN both have their own merits. However, when it comes to cross-subject situations, our method outperforms LE-ConvMN in all respects except inference. Our method have excellent performance in subject-specific situations and achieve state-of-the-art performance in cross-subject situations.

V. Discussion

BERT-based models were proposed to estimate finger joints from the sEMG signal and compared with LSTM, TCN and LE-ConvMN. CC and NRMSE were applied to measure the quality of estimation and average time cost per epoch, convergence time, and inference time were used to measure the efficiency. The results indicated that BERT-based models outperformed classical models among the 10 selected single subjects, but LE-ConvMN was still the state-of-the-art model in subject-specific situations. However, the new proposed method significantly outperformed all other models on multiple subjects simultaneously, that is, our method has stronger generalization ability.

In addition, BERT-based models are more efficient to train and faster to converge, as their structure allows models to train in parallel. But from the unique self-attention mechanism, models can seldom benefit in the inference efficiency of practical application. The strategy of hard sample mining and smooth layer lead to mild improvement. There is no further improvement on the Ninapro dataset, which may be due to the small amount of data in Ninapro DB2 being already well processed, including little noise and the feature is homogeneous in a single individual. However, when we applied GHM or OHEM to multiple subjects, due to the increase in the number of features of subjects, the number of hard samples increases, hard sample mining can lead to improvement in performance and the time cost is affordable. The smooth layer makes the estimation smoother and more stable, which is closer to reality.

Although there are methods based on transfer learning that allow models to adapt to different subjects, it is still difficult to find a model that can fit multiple subjects simultaneously. However, the proposed trained model cannot be applied to new subjects directly, unless the new subjects are involved in the training stage. That is, when we need the trained model to work on new subjects, the training data of the new subjects should be put together with that of the former subjects, and sometimes, it is time consuming. The more efficient way to extend our method to new subject is still transfer learning. As BERT is a high-quality pre-trained model itself [24], the BERT-based method can potentially contribute to transfer learning methods, and it would be our future work. At present, it is still a challenge to design a universal method to adapt to general individuals, our method is an important advance in this aspect.

There are several limits to our work. Only subjects with intact hands are selected in this paper, which leads to the lack of sufficient validation of the generality of our method. When choosing subjects, channels and movements, we deliberately avoided unreasonable or bad data caused by collecting errors in Ninapro and thus missing validation of model robustness. Inference time should be shortened to meet the need for practical applications. As for BERT-based structures, they are based on transformers and benefit from attention mechanisms, which leads to a higher delay in inference. Recently there has been lots of research about efficient transformers [31] to improve the efficiency of transformer-based structures, which can be the direction of subsequent improvement.

VI. Conclusion and Future Work

In this paper, we introduced a BERT-based method to estimate the continuous motions of the hands. To extract spatial and temporal information from sEMG signals, a BERT-based structure was designed to better meet the requirements in cross-subject situations in clinical use. μ-law normalization was also introduced to better utilize the hidden information of the small magnitude of sEMG. GHM and OHEM were applied in the training stage to better estimate hand motion from sEMG stably. Subsequently, two classical and a recent algorithms in continuous motion estimation from the sEMG signal were compared with our BERT-based method. The results showed that our method achieves considerable accuracy and stability on single subjects. For the first time, additionally, the proposed method can be applied to multiple different subjects simultaneously and outperformed the other models in this scenario and reached state-of-the-art results.

In future work, more subjects, movements and channels can be considered to evaluate the robustness and stability of the model. Although BERT is more efficient in training, the calculation of the full attention mechanism is time consuming in inferring. Since continuous motion estimation requires high efficiency, one can also improve the method with efficient transformers. Although GHM and OHEM were verified as mildly helpful in our work, as the unique mechanism of GHM and OHEM, there is potential for hard sample mining to work well when amounts of data increase in the future. It is expected that transformer-based, attention-based models and hard sample mining strategies contribute more and more to human-computer interaction and collaboration.

References

[1] P. D. Lawrence and W.-C. Lin, “Statistical decision making in the real-time control of an arm aid for the disabled,” *IEEE Trans. Syst., Man, Cybern.*, vols. SMC-2, no. 1, pp. 35–42, Jan. 1972.

[2] M. Ahsan, M. I. Ibrahimy, and O. O. Khalifa, "EMG signal classification for human computer interaction: A review," *Eur. J. Sci. Res.*, vol. 33, no. 3, pp. 480–501, 2009.

[3] I. Kapandji, "The physiology of the joints, volume I, upper limb," *Amer. J. Phys. Med. Rehabil.*, vol. 50, no. 2, p. 96, 1971.

[4] M. Längkvist, L. Karlsson, and A. Loutfi, "A review of unsupervised feature learning and deep learning for time-series modeling," *Pattern Recognit. Lett.*, vol. 42, pp. 11–24, Jun. 2014.

[5] M. Zanghieri, S. Benatti, A. Burrello, V. Kartsch, F. Conti, and L. Benini, "Robust real-time embedded EMG recognition framework using temporal convolutional networks on a multicore IoT processor," *IEEE Trans. Biomed. Circuits Syst.*, vol. 14, no. 2, pp. 244–256, Apr. 2020.

[6] E. Rahimian, S. Zabihi, A. Asif, D. Farina, S. F. Atashzar, and A. Mohammadi, "TEMGNET: Deep transformer-based decoding of upperlimb sEMG for hand gestures recognition," 2021, *arXiv:2109.12379*.

[7] L. Bi, A. G. Feleke, and C. Guan, "A review on EMG-based motor intention prediction of continuous human upper limb motion for human–robot collaboration," *Biomed. Signal Process. Control*, vol. 51, pp. 113–127, May 2019.

[8] S. Duprey, A. Naaim, F. Moissenet, M. Begon, and L. Chèze, "Kinematic models of the upper limb joints for multibody kinematics optimisation: An overview," *J. Biomech.*, vol. 62, pp. 87–94, Sep. 2017.

[9] M. Sartori, G. Durandau, S. Došen, and D. Farina, "Robust simultaneous myoelectric control of multiple degrees of freedom in wrist-hand prostheses by real-time neuromusculoskeletal modeling," *J. Neural Eng.*, vol. 15, no. 6, Dec. 2018, Art. no. 066026.

[10] L. Pan, D. L. Crouch, and H. Huang, "Myoelectric control based on a generic musculoskeletal model: Toward a multi-user neural-machine interface," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 7, pp. 1435–1442, Jul. 2018.

[11] Q. Zhang, R. Liu, W. Chen, and C. Xiong, "Simultaneous and continuous estimation of shoulder and elbow kinematics from surface EMG signals," *Frontiers Neurosci.*, vol. 11, p. 280, May 2017.

[12] F. Quivira, T. Koike-Akino, Y. Wang, and D. Erdogmus, "Translating sEMG signals to continuous hand poses using recurrent neural networks," in *Proc. IEEE EMBS Int. Conf. Biomed. Health Informat. (BHI)*, Mar. 2018, pp. 166–169.

[13] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2009.

[14] J. Fan et al., "Improving semg-based motion intention recognition for upper-limb amputees using transfer learning," *Neural Comput. Appl.*, early access, pp. 1–11, Jul. 2021.

[15] U. Côté-Allard et al., "Deep learning for electromyographic hand gesture signal classification using transfer learning," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 4, pp. 760–771, Apr. 2019.

[16] E. Rahimian, S. Zabihi, S. F. Atashzar, A. Asif, and A. Mohammadi, "XceptionTime: Independent time-window xceptiontime architecture for hand gesture classification," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 1304–1308.

[17] W. Guo et al., "Long exposure convolutional memory network for accurate estimation of finger kinematics from surface electromyographic signals," *J. Neural Eng.*, vol. 18, no. 2, 2021, Art. no. 026027.

[18] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[19] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *CoRR*, vol. abs/1803.01271, pp. 1–14, Mar. 2018.

[20] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[21] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. 4th Int. Conf. Learn. Represent. (ICLR)*, Y. Bengio and Y. LeCun, Eds., San Juan, PR, USA, May 2016, pp. 1–10.

[22] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 1–9.

[23] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, vol. 1. Stroudsburg, PA, USA: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186.

[24] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[25] *Pulse Code Modulation (PCM) of Voice Frequencies*, ITU, Geneva, Switzerland, 1988.

[26] E. Rahimian, S. Zabihi, A. Asif, D. Farina, S. F. Atashzar, and A. Mohammadi, "FS-HGR: Few-shot learning for hand gesture recognition via electromyography," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 1004–1015, 2021.

[27] A. Shrivastava, A. Gupta, and R. Girshick, "Training region-based object detectors with online hard example mining," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 761–769.

[28] B. Li, Y. Liu, and X. Wang, "Gradient harmonized single-stage detector," in *Proc. AAAI Conf. Artif. Intell.*, 2019, vol. 33, no. 1, pp. 8577–8584.

[29] M. Atzori et al., "Electromyography data for non-invasive naturally-controlled robotic hand prostheses," *Sci. Data*, vol. 1, no. 1, pp. 1–13, 2014.

[30] N. Ketkar and J. Moolayil, "Introduction to PyTorch," in *Deep Learning With Python*. Berkeley, CA, USA: Apress, 2021, pp. 27–91.

[31] Y. Tay, M. Dehghani, D. Bahri, and D. Metzler, "Efficient transformers: A survey," *ACM Comput. Surv.*, Apr. 2022.