

Long Exposure Convolutional Memory Network for accurate estimation of finger kinematics from surface electromyographic signals

Weiyu Guo^{2,3}, Chenfei Ma^{1,2}, Zheng Wang², Hang Zhang^{2,3}, Dario Farina⁴, Ning Jiang^{5*}, and Chuang Lin^{6**}

¹ College of Medicine and Biological Information Engineering, Northeastern University, 195 Chuangxin Road, Hunnan New District, Shenyang, 110016, P. R. China

² Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, 1068 Xueyuan Blvd., Xili Nanshan, Shenzhen, 518055, P. R. China

³ University of Chinese Academy of Sciences, Yanqihu Campus, 380 Huaibeizhuangcun, Huairou District, Beijing, 101408, P. R. China

⁴ Neuromechanics & Rehabilitation Technology Group, Department of Bioengineering, Faculty of Engineering, Imperial College London, London SW7 2AZ, UK

⁵ Department of Systems Design Engineering, University of Waterloo, 200 University Ave. W., Waterloo, Ontario, Canada, N2L 3G1

⁶ Dalian Maritime University, 1 Linghai Rode, Ganjingzi District, Dalian, 116026, P.R. China

Weiyu Guo and Chenfei Ma equally contributed to this work

*Corresponding author: ning.jiang@uwaterloo.ca (Ning Jiang)

** Principal corresponding author: linchuang_78@126.com (Chuang Lin)

Abstract

Objective. Estimation of finger kinematics is an important function of an intuitive human-machine interface, such as gesture recognition. Here, we propose a novel deep learning method, named Long Exposure Convolutional Memory Network (LE-ConvMN), and use it to proportionally estimate finger joint angles through surface electromyographic (sEMG) signals. **Approach.** We use a convolution structure to replace the neuron structure of traditional Long Short-Term Memory (LSTM) networks, and use the long exposure data structure which retains the spatial and temporal information of the electrodes as input. The Ninapro database, which contains continuous finger gestures and corresponding sEMG signals was used to verify the efficiency of the proposed deep learning method. The proposed method was compared with LSTM and Sparse Pseudo-input Gaussian Process (SPGP) on this database to predict the 10 main joint angles on the hand based on sEMG. The correlation coefficient (CC) was evaluated using the three methods on eight healthy subjects, and all the methods adopted the root mean square (RMS) features. **Main results.** The experimental results showed that the average CC, RMSE, NRMSE of the proposed LE-ConvMN method (0.82 ± 0.03 , 11.54 ± 1.89 , 0.12 ± 0.013) was significantly higher than SPGP (0.65 ± 0.05 , $p < 0.001$; 15.51 ± 2.82 , $p < 0.001$; 0.16 ± 0.01 , $p < 0.001$) and LSTM (0.64 ± 0.06 , $p < 0.001$; 14.77 ± 3.21 , $p < 0.001$; 0.15 ± 0.02 , $p < 0.001$). Furthermore, the proposed real-time-estimation method has a computation cost of only approximately 82 ms to output one state of ten joints (average value of 10 tests on TitanV GPU). **Significance.** The proposed LE-ConvMN method could efficiently estimate the continuous movement of fingers with sEMG, and its performance is significantly superior to two established deep learning methods.

Keywords: simultaneous, proportional, estimation, finger joint angle, surface electromyography, convolutional Long Short-Term Memory network

Introduction

Currently, robots are extensively used in industrial, military, commercial, and medical applications for improving the quality of human life. As such, the interactions between individuals and robots have become increasingly frequent. Human-robot interaction has become an integral part of systems such as active prostheses, robot-assisted surgery, drone reconnaissance. Building a user-friendly, intuitive and accurate human-machine collaboration (HMC) or human-computer interaction (HCI) has drawn much attention in recent years. One of the key challenges in HMC and HCI is the decoding of human movement intentions from biological signals. The surface electromyographic (sEMG) signal, electric manifestation of muscular contractions, is a direct source to extract effective information on movement intent and execution. This signal has been indeed used as a source for decoding human movement intentions for decades [1] [2].

The human hand is extremely complex [3], embedding a large number of afferents and intrinsic muscles that articulates >20 degrees of freedom (DoFs) [4]. Using sEMG, estimating intent of fine movements of the hand, such as different hand gestures, is a challenging task [5][6].

For decades, one of the most popular ideas in myoelectric analysis and control has been the use of manually selected features and machine learning algorithms to decode sEMG and recognize motions [7]. Recently, deep learning algorithms have been used that can automatically select the features, which is more effective for sEMG applications. Most of the related works on sEMG pattern recognition focus on the improvement of classification accuracy and the number of discriminated motions [3][8][9]. Phinyomark *et al.* [10] investigated fifty time-domain and frequency-domain features for classifying ten upper limb motions using sEMG. Krasoulis *et al.* [11] used the exp-kernel to solve the force regression and movement classification problems. Zanghieri *et al.* [8] proposed Temporal Convolutional Networks (TCNs) for robust gesture recognition. Although these classification algorithms are able to distinguish actions commonly used in daily life, they do not provide a natural and fine-grained control. More recent literature began to shift research focus from discrete movement classification to continuous movement regression. Although the estimation of continuous human motor intention from EMG signal and its application have been investigated [12][13][14][15][16], to the best of our knowledge, the current methods do not satisfy the criteria for accuracy, real-time performance and large number of estimated joint angles, to allow continuous hand gesture recognition.

EMG-based continuous motion estimation approaches can be categorized as model-based and model-free approaches. Model-based approaches include kinematics models [17], musculoskeletal models [18][19], or dynamic models in

general. However, the model-based approaches need to build a complex physical model and it is generally difficult to accurately estimate the relation between sEMG and joint angles. The model-based approaches are generally used to determine physical parameters such as position and acceleration. On the other hand, currently, researchers tend to use model-free approaches when facing practical control problem. For example, Lin *et al.* proposed a simultaneous and proportional multiple degree of freedom (DOF) myoelectric control method for active prostheses using sparse constraint non-negative matrix factorization. Zhang *et al.* [20] presented a simultaneous and continuous kinematics estimation method which used a single ANN for 4-DoFs across shoulder and elbow joints. Xiloyannis *et al.* [21] used the Gaussian process to estimate hand movements to achieve an accuracy of 0.79. Quivira *et al.* [22] proposed a hand pose estimation approach from low-cost sEMG systems using recurrent neural networks (RNN). However, the estimation accuracy and time delay of these methods are still below the needs for clinical translation.

In order to meet the needs on precision and low-latency, we propose a novel and low-computational-cost approach, which we named Long Exposure Convolutional Memory network (LE-ConvMN). This approach builds the connection between sEMG recordings and hand kinematics. This method was validated on the currently widely used Ninapro dataset and compared with two classical methods, *i.e.* LSTM [23] and Sparse Pseudo-input Gaussian Process (SPGP) [21]. These two compared methods are those with current best performance for multiple joint angle estimation of finger movements [24]. The experimental results showed that LE-ConvMN has the best estimation performance on both stability and accuracy.

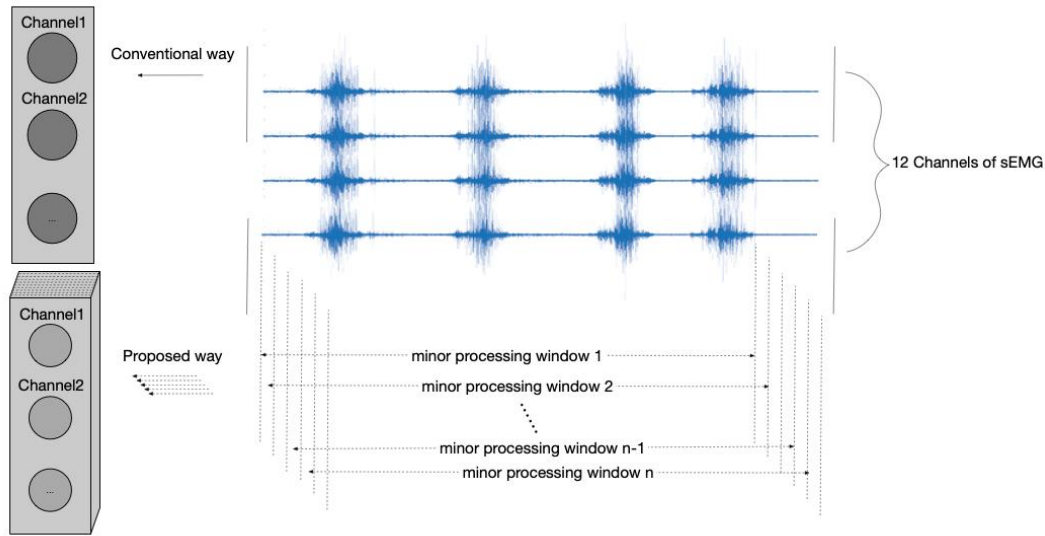


Fig. 1. One-step RMS to build 3D inputs which contain more detail information on time and space. The minor processing window contains 12 channels 100ms sEMG and the sliding step-length is 0.5 ms. sEMG is collected from 12 electrodes which 1–8 electrodes equally spaced around the forearm 9 and 10 spaced on the surface of flexor digitorum superficialis and digitorum superficial extenso, 11 and 12 measuring signals from Biceps Brachii and Triceps Brachii.

Method

Dataset

Ninapro [25] is currently the repository with the largest number of intact and hand amputated subjects in the literature. In this study, we design experiments on 6 movements shown in Fig. 3 from 8 subjects. Because the training of the deep learning algorithm is time consuming (e.g., LSTM took 33.25 hours), we choose 8 subjects out of the database. However, these 8 subjects are chosen to cover the all subjects' information as much as possible, i.g. the range of the height weights, gender, age and dominant hand are 154cm-187cm, 50kg-90kg, 5-male/3-female, 24-35, 6-right/2-left, respectively. Grasping movement is the most commonly used hand movement in daily life. We have selected these 6 types of grasping movement based on representative shapes and diameters. Hand kinematics was measured using a 22-sensor

CyberGlove II data-glove and sEMG is collected by Delsys Trigno Wireless System, which comprises 12 wireless sEMG electrodes. These electrodes sample the raw sEMG signal at a rate of 2 kHz. As shown in Fig. 2, we choose Proximal Interphalangeal point (PIP) and Metacarpophalangeal point (MCP) as estimated joint because they are the main active joints in the grasping movement.

The hand kinematics movement was collected at 20 Hz and resampled to 2000 Hz to synchronise with the sEMG signals. The sEMG signals and hand joint angle signals were divided into fragment sequences of 100-ms duration and sliding step-length of 0.5 ms. To reduce noise and simplify the computation, the root mean square (RMS) value was chosen as feature. After the RMS processing, each movement (containing six trials) was segmented into training dataset (four trials) and testing dataset to perform the train-test experiment on the LE-ConvMN as well as the two contrast neural networks with respect to the hand joint angle movements.

Parameters for evaluation

Pearson correlation coefficient is the most commonly used linear correlation coefficient, which can reflect the linear



Fig. 2. CyberGlove II data-glove. The red dot represents our estimate of 10 angles

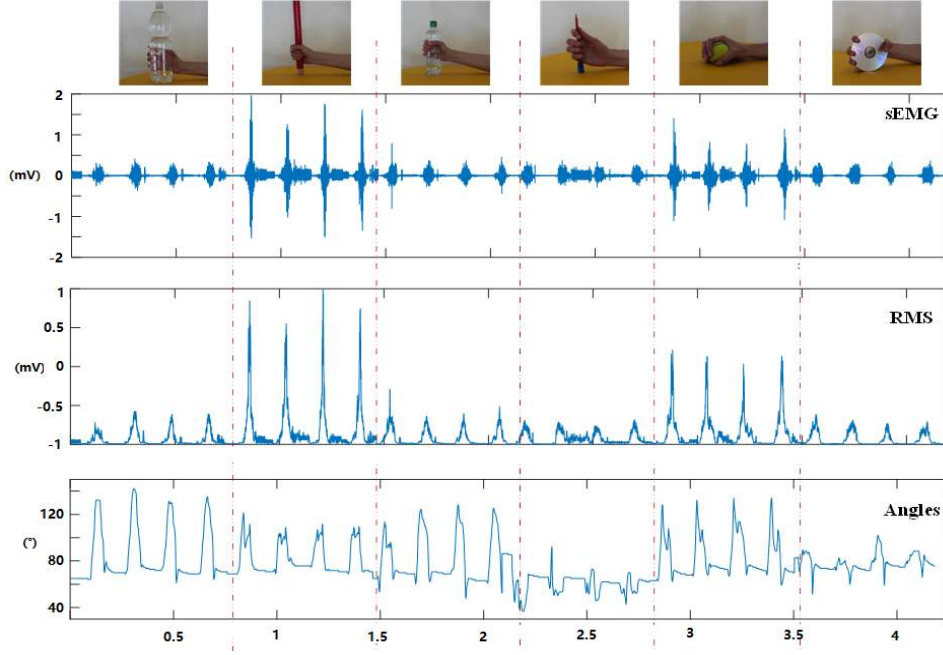


Fig. 3. One channel sEMG signal, RMS and joint angle after 6 types of action combinations

correlation between two variables. Here, we use it to measure the correlation between the estimated joint angle and the actual joint angle. Its calculation formula is as follows:

$$cc = \frac{\sum_{i=1}^N (\theta_{est} - \bar{\theta}_{est})(\theta_{real} - \bar{\theta}_{real})}{\sqrt{\sum_{i=1}^N (\theta_{est} - \bar{\theta}_{est})^2} \sqrt{\sum_{i=1}^N (\theta_{real} - \bar{\theta}_{real})^2}} \quad (1)$$

Where θ_{est} , $\bar{\theta}_{est}$, θ_{real} , $\bar{\theta}_{real}$ presents the value of estimated joint angle, the mean value of estimated joint angles, the value of real joint angle and the mean value of real joint angles, respectively. The CC value is between -1 and 1, which can be used to evaluate the performance of the algorithm. The closer CC value is to 1, the more similar the predicted finger trajectory is to that of the actual movement, and the higher accuracy of the estimation can reach.

Root mean square error (RMSE) can reflect the deviation between two variables. Here, RMSE is used to evaluate the deviation between the estimated value and the measured value of each joint angle, in degrees($^{\circ}$). RMSE is calculated as follows:

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (\theta_{est} - \theta_{real})^2}{N}} \quad (2)$$

RMSE is the numerical deviation between the predicted angle and the actual angle, so it can reflect the performance of the algorithm from the angle amplitude of joint. For the same joint, the smaller RMSE is, the closer the predicted value is to the actual value, the higher accuracy of the estimation can reach.

Due to the different range of motion of each joint, RMSE is unable to compare the performance of algorithms between different joints. So, we defined Normalized RMSE as a complement to evaluate different joints:

$$NRMSE = \frac{RMSE}{\theta_{MAX} - \theta_{MIN}}, \quad (3)$$

where, θ_{MAX} represents the maximum value of the actual measured angle of a joint, θ_{MIN} represents the minimum value of the actual measured angle of the joint.

Statistical Analysis

Due to the limited number of subjects, Friedman test was used to analyze significant differences among subjects with different evaluating methods, and Wilcoxon signed-rank test was used for post hoc comparisons. P value is corrected by Bonferroni correction.

Sparse Pseudo-input Gaussian Process (SPGP)

For non-linear non-parametric regression and classification, especially in high-dimensional space, the Gaussian process (GP) has shown to have an outstanding performance [21][26]. Assuming a dataset $\mathcal{D} = \{X, Y\}$, the input vector $X = \{x_n\}_{n=1}^N$ and the target value $Y = \{y_n\}_{n=1}^N$, we aim to find a function relating X and Y [1]. According to the standard GP, the targets are assumed to obey a Gaussian distribution $Y \sim \mathcal{N}(0, K_N + \sigma_n^2 I)$, where $\sigma_n^2 I$ present Gaussian noise with zero mean and σ_n^2 variance, and K_N is the covariance matrix commonly calculated by a kernel function, such as the commonly used

Radial Basis Function (RBF) kernel. When given a new point x , the distribution of the target value y is:

$$p(y|x, \mathcal{D}, \theta) = N\left(y \middle| \frac{k_x^T(K_N + \sigma_n^2 I)^{-1}y}{K_{xx} - k_x^T(K_N + \sigma_n^2 I)^{-1}k_x + \sigma^2}\right) \quad (4)$$

where $k_x = K(x_n, x)$ and $k_{xx} = K(x, x)$, and θ and σ^2 hyperparameters. Unfortunately, when the data set is large, the computational complexity and time of the standard GP would be too large for most practical applications. The complexity of training will be $\mathcal{O}(N^3)$ because the dimension of K is N^2 . As a result, the complexity of the testing will be at least $\mathcal{O}(N^2)$. In order to solve these computational problems, Snelson *et al.* proposed SPGP (Sparse Pseudo-input Gaussian Process) where the entire set N in standard GP is replaced by a small number of points M ($M \ll N$) [2]. Given the pseudo-input $\bar{X} = \{\bar{x}_m\}_{m=1}^M$, and the pseudo target $\bar{F} = \{\bar{f}_m\}_{m=1}^M$, the distribution of a single point is:

$$p\{y|x, \bar{X}, \bar{F}\} = \mathcal{N}(y|k_x^T K_M^{-1} \bar{F}, K_{xx} - k_x^T K_M^{-1} k_x + \sigma^2) \quad (5)$$

where $K_M = K(\bar{x}_m, \bar{x}_m)$, and $k_M = K(\bar{x}_m, x)$.

As for the entire data, the complete likelihood is given as:

$$p\{Y|X, \bar{X}, \bar{F}\} = \mathcal{N}(Y|K_{NM} K_M^{-1} \bar{F}, \Lambda + \sigma^2 I) \quad (6)$$

where $\Lambda = \text{diag}(\lambda)$, $\lambda_n = K_{nn} - k_n^T K_M^{-1} k_n$, and $K_{NM} = K(x_n, \bar{x}_m)$. Then the posterior distribution of \bar{F} can be found as:

where $Q_M = K_M + K_{MN}(\Lambda + \sigma^2 I)^{-1} K_{NM}$. Now given a new input x' , the distribution of the target y' is:

$$\begin{aligned} p(y'|x', \mathcal{D}, \bar{X}) &= \int d\bar{F} p(y'|x', \bar{X}, \bar{F}) p(\bar{F}|\mathcal{D}, \bar{X}) \\ &= \mathcal{N}(y'|\mu', \sigma'^2) \end{aligned} \quad (8)$$

where $\mu' = k'^T Q_M^{-1} K_{MN}(\Lambda + \sigma^2 I)^{-1} Y$ and $\sigma'^2 = k'^T (K_M^{-1} - Q_M^{-1}) k' + \sigma^2$.

The complexity of Q_M is $\mathcal{O}(M^2 N)$ and therefore the complexity in the testing process is $\mathcal{O}(M^2)$. Thus, the complexity and computation time have no association with the size of the data set. In the current work, we utilized the code from the Sheffield Machine Learning Group [2].

Long short-term memory (LSTM)

Long short-term memory (LSTM) neural network is a type of recurrent neural network (RNN). The basic recurrent circle provides RNN the capacity to bring in the information from previous iterations. Therefore, RNN has been used widely in solving problems with temporal information [27]. With the growth of data complexity, the short recurrent circle could not meet the demand which led to the proposal of LSTM. The LSTM structure is built to implement long term dependencies. Being one type of RNN, LSTM has been proved more capable on the regression problem or continuous output fitting problem. It controls the information pass from time t to time $t+1$ by a gate structure that contains a matrix weight. Hence, important information maintains a relatively long term to take effect. LSTM has been employed for natural language processing [28], intention recognition [29], continuous control [24], and many other applications. In previous studies [22][30], RNN can extract kinematics information efficiently from sEMG signals. In this study, we take advantages of LSTM to implement simultaneous and proportional control as a comparison to our proposed method. The procedure for the LSTM algorithm was implemented as follows.

Assuming $X = [x_0, x_1, \dots, x_t, \dots]$ is the one-dimensional input vector, the middle-hidden layer state vector $H = [h_0, h_1, \dots, h_t, \dots]$ can be written as:

$$H = \partial(W_i x_t + W_h h_{t-1} + b_h) \quad (9)$$

where W_i stands for the input weight matrix, W_h is the hidden weight matrix, b_h presents the hidden bias matrix and ∂ stands for the activation function. Then, the output sequence $Y = [y_0, y_1, \dots, y_t, \dots]$ is obtained as:

$$Y = \rho(W_o h_t + b_o) \quad (10)$$

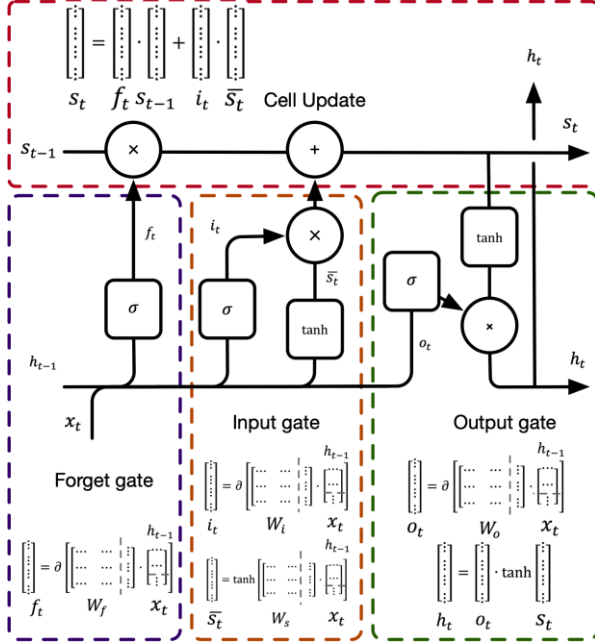


Fig. 4. The structure of a LSTM cell. where i_t, o_t, f_t, s_t stand for input gate, output gate, forget gate and the LSTM cell state. \tilde{s} stand for alternate cell state.

$$p(\bar{F}|\mathcal{D}, \bar{X}) \mathcal{N}(\bar{F}|K_M Q_M^{-1} K_{MN}(\Lambda + \sigma^2)^{-1} Y, K_M Q_M^{-1} K_M) \quad (7)$$

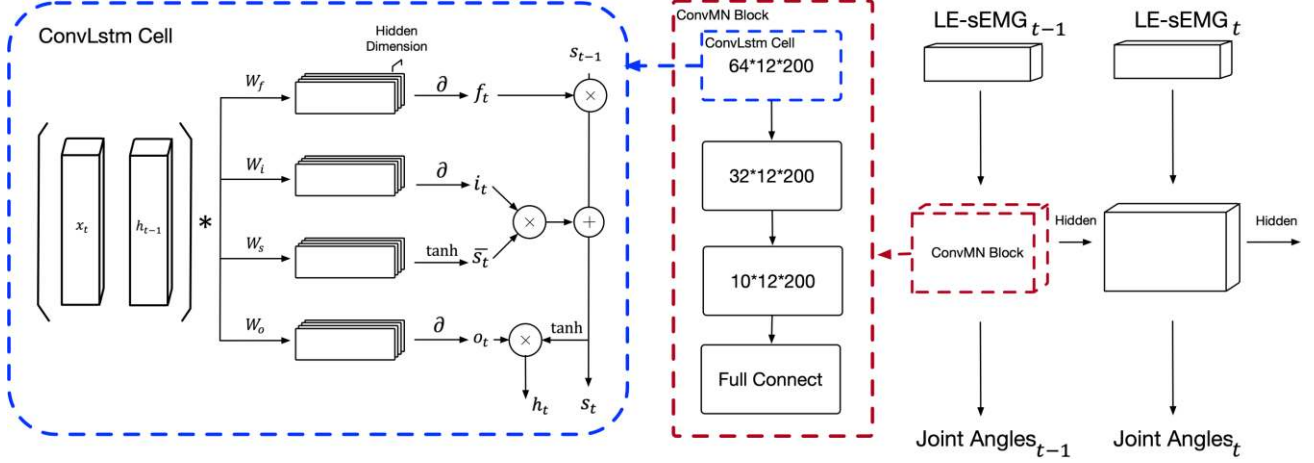


Fig. 5. ConvLstm cell uses convolution to achieve the calculation of each gate and the number of convolution channels is equal to the number of hidden states. The Structure of ConvMN Block consists of three ConvLstm layers with hidden state 64, 32, 10 and one fully connected layer. Inside the ConvMN Block, the hidden state is delivered layer by layer and the dimensions are continuously reduced, and finally the fully connected layer does feature mapping. The inputs of ConvMN block at time t consists of the LE-sEMG at time t and the hidden state of the third layers of the ConvMN block at time $t-1$ ($10 * 12 * 200$). Hidden state is computed by forget gate, input gate and output gate.

Similarly, W_o presents the output weight matrix, b_o is the output bias and ρ stands for the activation function. Weights in the LSTM control the maintenance of information. LSTM includes input gate, forget gate, memory cell and output gate. The structure can be seen in Fig. 4. A single iteration is expressed as follows:

$$i_t = \partial(W_{xi}x_t + W_{ui}h_{t-1} + b_i) \quad (11)$$

$$\bar{s} = \tanh(W_{xs}x_t + W_{us}h_{t-1} + b_s) \quad (12)$$

$$o_t = \partial(W_{xo}x_t + W_{uo}h_{t-1} + b_o) \quad (13)$$

$$f_t = \partial(W_{xf}x_t + W_{uf}h_{t-1} + b_f) \quad (14)$$

$$s_t = f_t \cdot s_{t-1} + i_t \cdot \bar{s} \quad (15)$$

$$h_t = o_t \cdot \rho(s_t) \quad (16)$$

where i_t, o_t, f_t, s_t stand for input gate, output gate, forget gate and the LSTM cell state (which influences the output at the next iteration). The W terms stand for the weight matrices and their subscripts denote their respective positions (e.g. W_{xi} denotes the weight matrix from the input gate to the input), the b terms stand for bias vectors (b_i is the input gate bias vector). ∂ and ρ represent the activation functions of the hidden layer and the output layer, respectively. The symbol ' \cdot ' represents element-wise product of the vectors. The three gates are influenced by the last step hidden state and the cell state. Moreover, the cell state is controlled by the forget gate from the last step.

Long Exposure-convolutional Memory Network (LE-ConvMN)

Although LSTM has been proven to have powerful ability to process temporal data, it has suboptimal performance on spatial data and depends on many parameters. The major drawback of LSTM in handling spatiotemporal data is that it

has to flatten the multi-dimensional inputs into a 1D vector before processing and, as a result, all the spatial information embedded in the original input data is lost. To address this problem, Shi *et al.* proposed a new network which has convolutional structures in both the input-to-state and state-to-state transitions, called ConvLstm, and used it for precipitation nowcasting [31]. The key equations of ConvLstm are shown in below:

$$i_t = \partial(W_{xi} * x_t + W_{ui} * h_{t-1} + b_i) \quad (17)$$

$$\bar{s} = \tanh(W_{xs} * x_t + W_{us} * h_{t-1} + b_s) \quad (18)$$

$$o_t = \partial(W_{xo} * x_t + W_{uo} * h_{t-1} + b_o) \quad (19)$$

$$f_t = \partial(W_{xf} * x_t + W_{uf} * h_{t-1} + b_f) \quad (20)$$

$$s_t = f_t \cdot s_{t-1} + i_t \cdot \bar{s} \quad (21)$$

$$h_t = o_t \cdot \rho(s_t) \quad (22)$$

The advantage of ConvLstm is that the internal state can maintain the time and space dimension of the input data. ConvLstm has also good performance on temporal-spatial data. Thus, we propose an extension of ConvLstm, namely Long Exposure-convolutional Memory Network (LE-ConvMN) which is more suitable for processing myoelectric data. The LE-ConvMN determines the future state of a certain cell in the grid by the inputs and past states of its local neighbours. This can be achieved by using a convolution operator in the state-to-state and input-to-state transitions. The differences between the adjacent inputs are not so large in this application, so the full-connection structure in LSTM is not necessary. Full connection will greatly increase the complexity of the model, reduce the training speed and generalization ability. The use of convolutional structures not

only reduces redundancy but also extracts combined information from multiple channels more efficiently (two-dimensional convolution is adopted).

In previous work, researchers used to extract features in time and frequency domain within each channel. However, the interaction between channels was less explored. As shown in Fig. 1, we convert the raw sEMG into a multi-channel Long exposure EMG sample collection which contains 100 sub-frames root mean square (RMS) features (1 sub-frame RMS feature is calculated from 100-ms EMG and the sliding step-length is 0.5 ms) rather than using 200 ms EMG signal as one features input. Obviously, comparing to other methods which only takes 1 frame feature input, the long exposure structure can make better usage of the information of EMG in time and space dimensions. Considering the requirement of data sufficiency for deep learning, we aim at a fine balance between presenting information to the network and the real-time performance of the pipeline. This operation is analogous to a photographer getting a richer photo by increasing the exposure times of the camera. Due to the enlargement of feature extraction area, we set the LE-ConvMN layer hidden sizes as 64, 32, 10 to compress volume and extract features in higher dimension. All convolution operations use 3×3 convolution kernels. The stride and padding are 1. Meanwhile, compressing the feature dimension can also effectively prevent overfitting. At the end of the network we added a fully connected layer to reorganize the mapping of features to the target space. The structure of LE-ConvMN is shown in Fig. 5.

When Long exposure EMG collection (100 sub-frames) is provided as input to LE-ConvMN, features in time and space dimension of EMG sub-frame are extracted in each layer and kept in hidden states until 100 sub-frames of data are extracted. Then the hidden states of this layer will transfer to the next layer. According to our design, the dimension of the hidden state is reduced (64-32-10) layer by layer, which can complete the transition of data from low-dimensional features to high-dimensional features. Finally, the fully connected layer will map the high-dimensional features to joint angles. Compared with traditional machine learning methods, LE-ConvMN can better perform the joint training of feature extraction and feature-to-target space mapping. In this study, we verify that adding a full connection layer at the end of the network can make the motion curve smoother.

Result

We built LSTM and LE-ConvMN on the Pytorch framework [32], and built SPGP on MATLAB, comparing their performance on continuous proportional estimation of finger movements.

LSTM and LE-ConvMN were trained on the same GPUs (Titan V). Both models converged after training for around

300 epochs and the average convergence time of LSTM and LE-ConvMN is 33.25 hours and 3.5 hours, respectively.

Fig. 6 shows the performance of the three contrast models for the eight experimental subjects and ten channels. We firstly performed Friedman test on the performance (RMSE, NRMSE, CC) of the three methods on the subjects and the channels, and the p-values were all less than 0.05, indicating that the performance of the three methods is significantly different. Afterwards, we performed Wilcoxon signed-rank test on (LE-ConvMN, SPGP) and (LE-ConvMN, LSTM). The p-values on subject-to-subject and joint-to-joint were both less than 0.05, indicating that LE-ConvMN significantly outperformed the other two methods. The average CC, RMSE, NRMSE of the proposed method (0.82 ± 0.03 , 11.54 ± 1.89 , 0.12 ± 0.01) was significantly higher than SPGP (0.66 ± 0.05 , $p=0.012$; 15.51 ± 2.81 , $p=0.025$; 0.16 ± 0.01 , $p=0.017$) and LSTM (0.64 ± 0.06 , $p=0.012$; 14.77 ± 3.21 , $p=0.025$; 0.15 ± 0.02 , $p=0.025$). In our previous work [24], we proved that LSTM has high estimation accuracy in a single class of actions, which is consistent with the results of the current study. However, the accuracy of LSTM from our current results is greatly reduced because of the combination of multiple motions, while LE-ConvMN maintained high accuracy. This indicates that our proposed network has stronger generalization ability in intra-class continuous estimation of finger movements than LSTM. Almost all of the LE-ConvMN estimation accuracies among all subjects are greater than 0.8, which indicates that the proposed LE-ConvMN has the capability of generalizing for most subjects. It is evident from Fig. 6(a) that the performance of LSTM and SPGP was particularly worse than LE-ConvMN on subjects 2, 3, 5. As for the RMSE and NRMSE, LE-ConvMN presents superior performance, especially for subject 3.

The estimation accuracy with each individual channel is shown in Fig. 6(b)(d)(f). LE-ConvMN always has the best performance for all ten channels. In Fig. 6(b), the LE-ConvMN surpasses LSTM and SPGP significantly in accuracies by 10% among all joint angles. Fig. 6(d) shows that the RMSE of LE-ConvMN is lower than LSTM and SPGP on each joint angle. As shown in Fig. 6(f), NRMSE also shows superior performance when applying LE-ConvMN. LE-ConvMN also shows high accuracy and stability for channels distinctly more difficult to predict (e.g. channel 6, 9, 13, 16, 17). Especially on joint angle 9, LE-ConvMN provides high accuracies, small variance, low RMSE and low NRMSE while LSTM and SPGP perform distinctly worse for this joint angle than for the others. This suggests that LE-ConvMN has the capability, stability and versatility to be applied on various subjects and joint angles.

In order to verify the effect of the noise may appear in practice, we add gaussian noise with SNR 10dB to the test data. Experimental results show that the noise lower the accuracy of the LE-ConvMN by 0.3% which proved that the method is

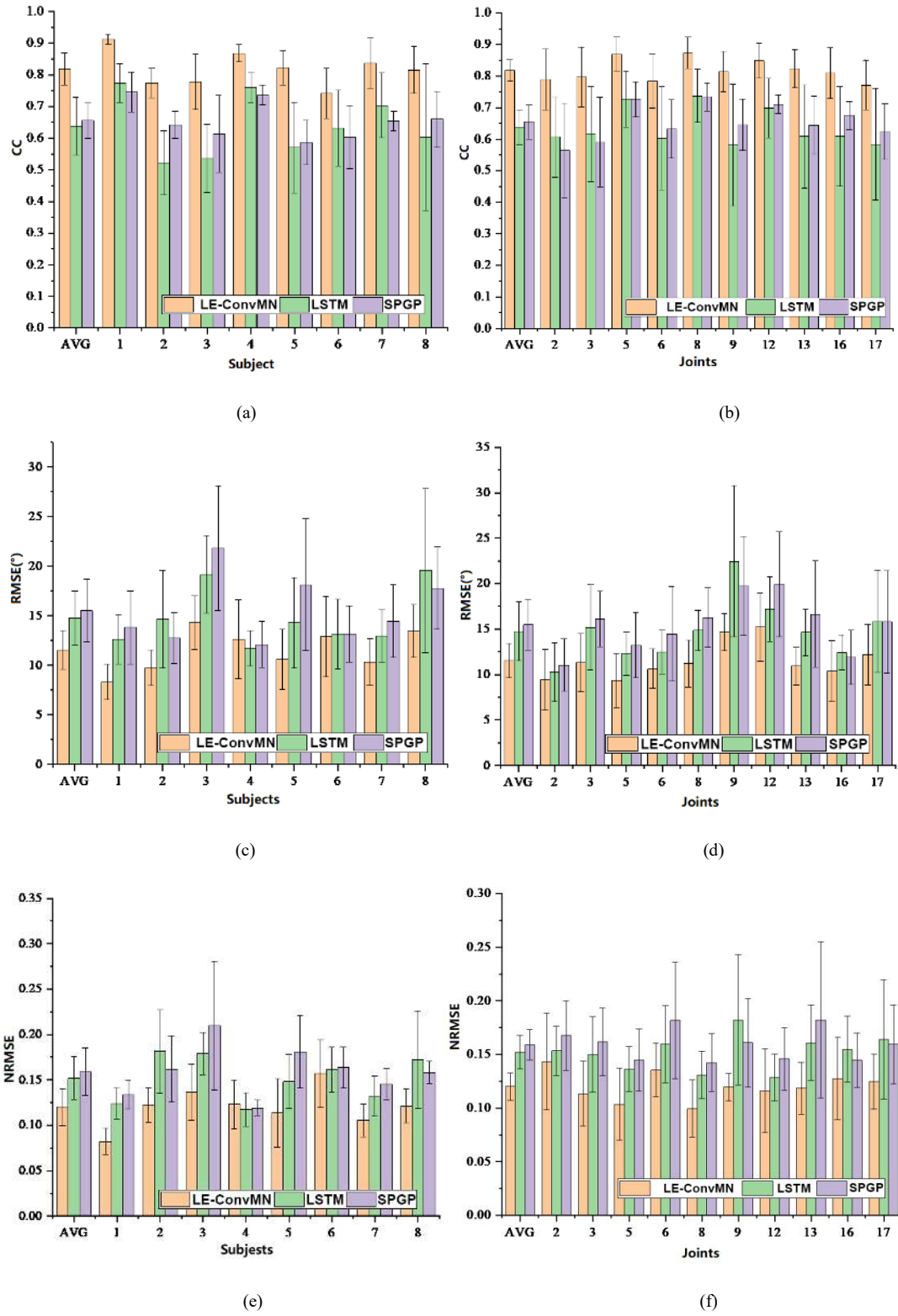
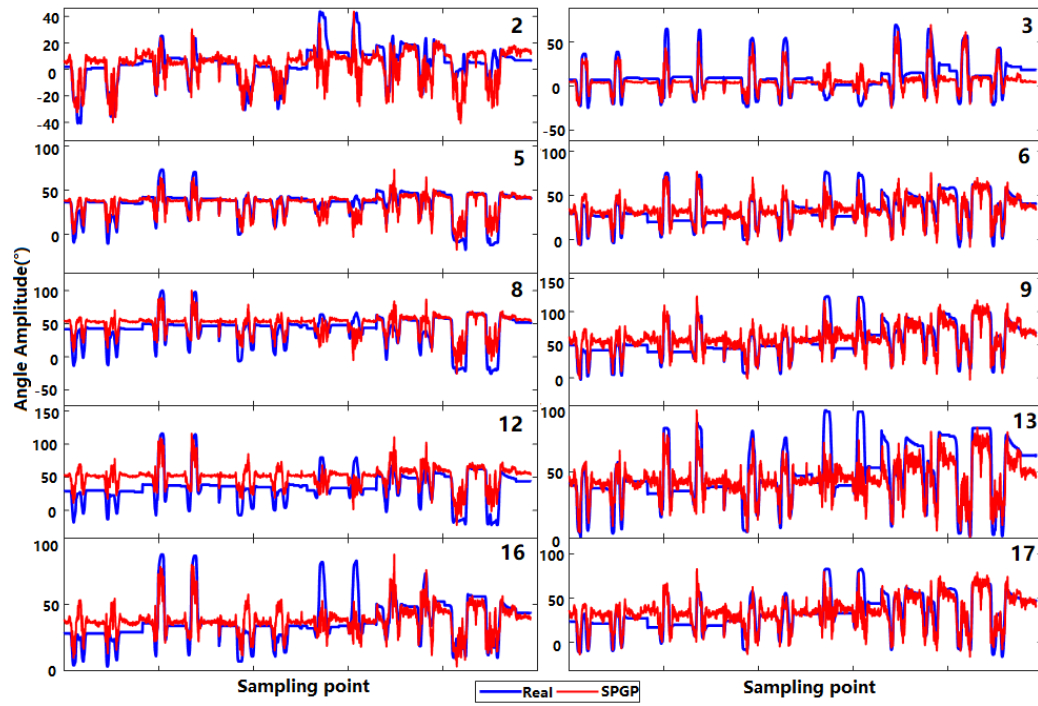
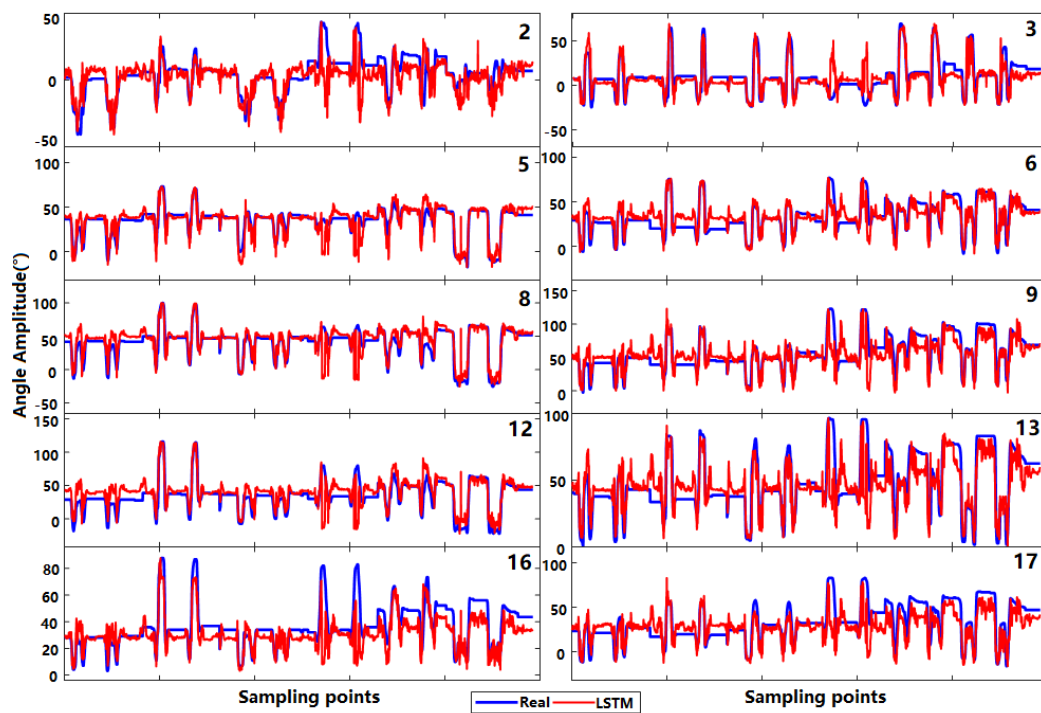


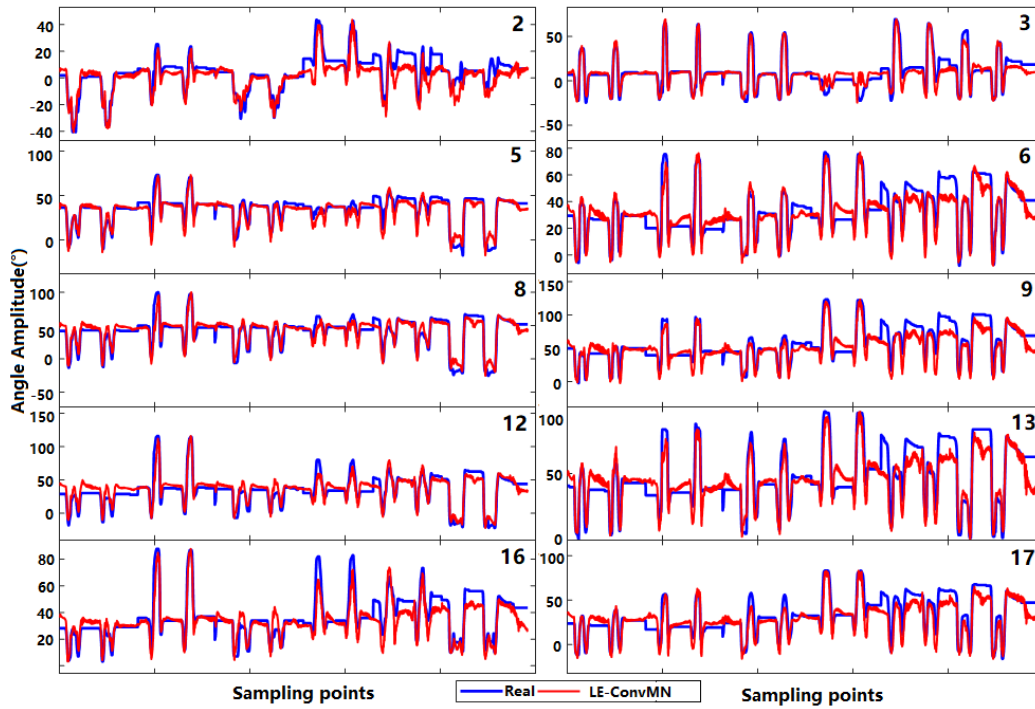
Fig. 6. Summary of the performance measures of 8 able-bodied subjects. From top to down, are the CC, RMSE, NRMSE, and overshoots indicates the variances.



(a)



(b)



(c)

Fig. 7. (a) (b) (c) shows the predicted and actual values of SPGP, LSTM and LE-ConvMN respectively. 8 subjects, each subject execute 6 grasp movements.

robust to this kind of noise.

The estimated joint angles with the three contrast models are plotted against the measured curves of participant 1 in Fig. 7. As is shown in Fig. 7(a), the SPGP model provides accurate trend but the amplitude is incorrect, which leads to the worst performance among the three models. On the other hand, Fig. 7 (b) shows that the predicted curve from the LSTM model estimates the correct trend but presents large error at turning points. Fig. 7 (c) shows that the predicted curve by LE-ConvMN is the most similar to the measured curve without distortion. As for the processing delay, the average delay of LE-ConvMN is 82ms (input step 50ms + calculation time 32ms.) which is well below the need for real-time requirement in myoelectric control (200 ms [33])

Discussion

The finger motion detection and prediction would greatly influence the performance of an HCI involving hand movements, such as gestures. In this study, the proposed approach of LE-ConvMN was compared with traditional mathematical (SPGP) and machine learning methods (LSTM), when finger movements are estimated from sEMG.

To accurately describe the estimation performance of each algorithm, correlation coefficients of each algorithm were measured between the actual and predicted finger joint angle. The results shown in Fig. 7 indicate that, among these 8

individual participants, the proposed method not only has significantly higher estimation accuracy but also provides better stability than SPGP and LSTM. Assuming applied in real-world application scenario, it would be difficult for SPGP and LSTM to identify the joint angles because of the heterogeneity of participants (LSTM performs better than SPGP on some participants only). In comparison, the proposed method has the ability to generalize and was superior to both LSTM and SPGP in all participants. As shown in Fig. 7, LSTM with its unique recurrent structure performs better than SPGP on most joint angles, LE-ConvMN performs the best on each angle due to the powerful structure pattern recognition ability of convolutional kernel, the extraordinary sequence data decoding capability of LSTM, and the specific long exposure data input method.

The advantage of the proposed estimation approach is further reflected in Fig. 7. The predicted joint angle curves of SPGP have a relatively long lag in response, especially during intervals of sudden inflections. The LSTM is able to detect the trend of the actual finger motion clearly but there are discrepancies in amplitude, especially when the joint angle is constant. Comparatively, the LE-ConvMN provides more continuous, stable and accurate joint angle curves to match the actual curves.

At present, the application of deep learning in the field of myoelectric control mostly relies on discrete estimation [8]

[9] [34] [35]. Although TCN can achieve good results in discrete estimation [8], it cannot be effectively applied to continuous estimates.

Researchers are trying to find effective methods to estimate the continuous movement of the hand. For example, Xiloyannis *et al.* proposed Gaussian Process Autoregression (GPA) for decoding neural information that enables a user to independently control 11 joints of the hand [36]. Ghaderi *et al.* applied Generalized Regression Neural Network (GRNN), a non-linear identification approach, to estimate finger kinematics (15 Degrees of Freedom) from sEMG signals [37]. Comparing with the two methods, LE-ConvMN can estimate more complex actions and provides higher accuracy. Li *et al.* proposed a time-delay recurrent neural network (TDRNN) method by using sEMG signals to predict the kinematics of hand and wrist but only applied it for estimates of 3 degrees of freedom. Compared with our previous work [24], LE-ConvMN does not need to build different models for different actions and has a faster training speed.

However, there are some limits in this work, such as the finger motions in this study mainly focused on the basic functional grasp motions. We have not tested the performance of LE-ConvMN on predicting other types of finger motions (e.g., "OK", "three", "V", and pinching). Therefore, the compatibility of the method with other types of finger motions remains to be tested. It has been mathematically proved that multilayer neural network can fit arbitrary functions [38], so it is possible to develop a deep learning network to meet the more complexed finger motions, we are considering to integrate more complex finger motions in the future work. Meanwhile, the deviation between the predicted and actual joint angle might be reduced by integrating adaptive type algorithms. Although off-line analysis is the starting point of exploration, there are still some issues (i.e., shift in electrode position, condition, and arm posture) to be solved when apply the method in practice. We have tried to solve the problem of electrode shifting or noise. In our former work, we have used high density electrodes array to simulate electrode shifting, and our preliminary work has been published [39][40]. But in this paper, the Ninapro database uses sparse electrodes, the distance between two sparse electrodes is too far, which is not so easy to simulate the electrode shifting like the actual situation. We have simulated moving the electrode in a large range (equal to the interval width of two neighboring sparse electrodes). This will obviously reduce the accuracy of the LE-ConvMN (more than 10%). In the future work, we will collect the shifted electrode data by our own equipment, which may better evaluate the robustness of the algorithm when meeting electrode shifting.

Conclusion

In this paper, we propose a LE-ConvMN-based deep learning model to estimate finger motion simultaneously and proportionally. According to the temporal and spatial characteristics of sEMG data and the real-time requirements of practical use, we designed a multi-channel long exposure input and convolution structure to reproduce the gate operation of LSTM. Then we chose two representative algorithms (SPGP and LSTM) for the continuous estimation field to compare them with LE-ConvMN. The experimental results showed that LE-ConvMN achieves higher accuracy, stability and smaller delay than LSTM, and is expected to play an important role in human-computer interaction and human-computer cooperation.

Acknowledgements

The work is supported by the Shenzhen Basic Research Grants (#JCYJ20170413152804728, #JCYJ20180507182508857), Pioneer Hundred Talents Program of Chinese Academy of Sciences (2017).

References

- [1] R. Reiter, "Eine neue elektrokunsthands," *Grenzgeb. Med.*, vol. 4, pp. 133–135, 1948.
- [2] M. R. 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] S. Micera, J. Carpaneto, and S. Raspopovic, "Control of Hand Prostheses Using Peripheral Information," *IEEE Rev. Biomed. Eng.*, vol. 3, pp. 48–68, 2010.
- [4] I. A. Kapandji, "The Physiology of the Joints. Vol. 1, Upper Limb," *Postgrad. Med. J.*, vol. 47, no. 544, pp. 140–140, 1971.
- [5] M. Hioki and H. Kawasaki, "Estimation of finger joint angles from sEMG using a recurrent neural network with time-delayed input vectors," in *IEEE International Conference on Rehabilitation Robotics*, 2009.
- [6] W. Batayneh, E. Abdulhay, and M. Allothman, "Prediction of the performance of artificial neural networks in mapping sEMG to finger joint angles via signal pre-investigation techniques," *Heliyon*, vol. 6, no. 4, p. e03669, 2020.
- [7] 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, no. 1, pp. 11–24, 2014.
- [8] 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, 2020.
- [9] 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, 2019.
- [10] A. Phinyomark, F. Quaine, S. Charbonnier, C. Serviere, F. Tarpin-Bernard, and Y. Laurillau, “EMG feature evaluation for improving myoelectric pattern recognition robustness,” *Expert Syst. Appl.*, vol. 40, no. 12, pp. 4832–4840, 2013.
- [11] A. Krasoulis, S. Vijayakumar, and K. Nazarpour, “Evaluation of regression methods for the continuous decoding of finger movement from surface EMG and accelerometry,” *Int. IEEE/EMBS Conf. Neural Eng. NER*, vol. 2015-July, pp. 631–634, 2015.
- [12] N. Jiang, S. Dosen, K. R. Muller, and D. Farina, “Myoelectric control of artificial limbs: is there a need to change focus? [In the Spotlight],” *IEEE Signal Process. Mag.*, vol. 29, no. 5, pp. 148–152, 2012.
- [13] T. Kapelner *et al.*, “Predicting wrist kinematics from motor unit discharge timings for the control of active prostheses,” *J. Neuroeng. Rehabil.*, vol. 16, no. 1, pp. 1–11, 2019.
- [14] L. Bi, A. Eleke, 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, 2019.
- [15] D. Farina *et al.*, “The extraction of neural information from the surface EMG for the control of upper-limb prostheses: emerging avenues and challenges,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 22, no. 4, pp. 797–809, 2014.
- [16] N. Jiang, K. B. Englehart, and P. A. Parker, “Extracting simultaneous and proportional neural control information for multiple-dof prostheses from the surface electromyographic signal,” *IEEE Trans. Biomed. Eng.*, vol. 56, no. 4, pp. 1070–1080, 2009.
- [17] 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, no. December, pp. 87–94, 2017.
- [18] 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, p. 066026 (15pp), 2018.
- [19] 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, 2018.
- [20] Q. Zhang, R. Liu, W. Chen, and C. Xiong, “Simultaneous and continuous estimation of shoulder and elbow kinematics from surface EMG signals,” *Front. Neurosci.*, vol. 11, no. MAY, pp. 1–12, 2017.
- [21] M. Xiloyannis, C. Gavriel, A. A. C. Thomik, and A. A. Faisal, “Gaussian Process Autoregression for Simultaneous Proportional Multi-Modal,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 10, pp. 1785–1801, 2017.
- [22] F. Quivira, T. Koike-Akino, Y. Wang, and D. Erdogmus, “Translating sEMG signals to continuous hand poses using recurrent neural networks,” *2018 IEEE EMBS Int. Conf. Biomed. Heal. Informatics, BHI 2018*, vol. 2018-Janua, no. March, pp. 166–169, 2018.
- [23] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [24] C. Wang, W. Guo, H. Zhang, L. Guo, C. Huang, and C. Lin, “sEMG-based continuous estimation of grasp movements by long-short term memory network,” *Biomed. Signal Process. Control*, vol. 59, p. 101774, 2020.
- [25] M. Atzori *et al.*, “Electromyography data for robotic hand prostheses,” pp. 1–13, 2014.
- [26] J. G. Ngeo, T. Tamei, and T. Shibata, “Continuous and simultaneous estimation of finger kinematics using inputs from an EMG-to-muscle activation model,” *J. Neuroeng. Rehabil.*, vol. 11, no. 1, pp. 1–14, 2014.
- [27] G. Williams, R. Baxter, H. He, S. Hawkins, and L. Gu, “A comparative study of RNN for outlier detection in data mining,” *Proc. - IEEE Int. Conf. Data Mining, ICDM*, no. May 2014, pp. 709–712, 2002.
- [28] M. Sundermeyer, R. Schlüter, and H. Ney, “LSTM neural networks for language modeling,” in *Thirteenth annual conference of the international speech communication association*, 2012.
- [29] W. Chen *et al.*, “EEG-based motion intention recognition via multi-task RNNs,” *SIAM Int. Conf. Data Mining, SDM 2018*, pp. 279–287, 2018.
- [30] P. Xia, J. Hu, and Y. Peng, “EMG-Based Estimation of Limb Movement Using Deep Learning With Recurrent Convolutional Neural Networks,” *Artif. Organs*, vol. 00, no. 00, 2017.
- [31] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W. Wong, and W. Woo, “Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting,” no. June, 2015.
- [32] N. Ketkar, “Introduction to PyTorch BT - Deep Learning with Python: A Hands-on Introduction,” N. Ketkar, Ed. Berkeley, CA: Apress, 2017, pp. 195–208.

- [33] K. Englehart and B. Hudgins, "A Robust, Real-Time Control Scheme for Multifunction Myoelectric Control," *IEEE Trans. Biomed. Eng.*, vol. 50, no. 7, pp. 848–854, 2003.
- [34] S. Tam, M. Boukadoum, A. Campeau-Lecours, and B. Gosselin, "A Fully Embedded Adaptive Real-Time Hand Gesture Classifier Leveraging HD-sEMG and Deep Learning," *IEEE Trans. Biomed. Circuits and Syst.*, vol. 14, no. 2, pp. 232–243, 2020.
- [35] W. Sun, H. Liu, R. Tang, Y. Lang, J. He, and Q. Huang, "sEMG-based hand-gesture classification using a generative flow model," *Sensors (Switzerland)*, vol. 19, no. 8, 2019.
- [36] M. Xiloyannis, C. Gavriel, A. A. C. Thomik, and A. A. Faisal, "Gaussian Process Autoregression for Simultaneous Proportional Multi-Modal Prosthetic Control with Natural Hand Kinematics," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 10, pp. 1785–1801, 2017.
- [37] P. Ghaderi, S. Karimimehr, M. E. Andani, and H. R. Marateb, "Hand kinematics estimation to control prosthetic devices: A nonlinear approach for simultaneous and proportional estimation of 15 DoFs," *2015 22nd Iran. Conf. Biomed. Eng. ICBME 2015*, pp. 233–238, 2016.
- [38] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, no. 5, pp. 359–366, 1989.
- [39] A. Stango, F. Negro, and D. Farina, "Spatial Correlation of High Density EMG Signals Provides Features Robust to Electrode Number and Shift in Pattern Recognition for Myocontrol," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 23, no. 2, pp. 189–198, 2015.
- [40] H. Zhang, C. Wang, W. Gou, L. Guo, and C. Lin, "DFNN-based gesture recognition with the shift and damage of the HD-sEMG electrodes," *IEEE Int. Conf. Robot. Biomimetics, ROBIO 2019*, no. December, pp. 1275–1279, 2019.