LSTM-AE for Domain Shift Quantification in Cross-Day Upper-Limb Motion Estimation Using Surface Electromyography

Tianzhe Bao, Chao Wang, Pengfei Yang, *Member, IEEE*, Sheng Quan Xie, *Senior Member, IEEE*, Zhi-Qiang Zhang, *Member, IEEE*, and Ping Zhou, *Senior Member, IEEE*

***Abstract*— Although deep learning (DL) techniques have been extensively researched in upper-limb myoelectric control, system robustness in cross-day applications is still very limited. This is largely caused by non-stable and time-varying properties of surface electromyography (sEMG) signals, resulting in domain shift impacts on DL models. To this end, a reconstruction-based method is proposed for domain shift quantification. Herein, a prevalent hybrid framework that combines a convolutional neural network (CNN) and a long short-term memory network (LSTM), i.e. CNN-LSTM, is selected as the backbone. The paring of auto-encoder (AE) and LSTM, abbreviated as LSTM-AE, is proposed to reconstruct CNN features. Based on reconstruction errors (RErrors) of LSTM-AE, domain shift impacts on CNN-LSTM can be quantified. For a thorough investigation, experiments were conducted in both hand gesture classification and wrist kinematics regression, where sEMG data were both collected in multi-days. Experiment results illustrate that, when the estimation accuracy degrades substantially in between-day testing sets, RErrors increase accordingly and can be distinct from those obtained in within-day datasets. According to data analysis, CNN-LSTM classification/regression outcomes are strongly associated with LSTM-AE errors. The average Pearson correlation coefficients could reach −0.986 ± 0.014 and −0.992 ± 0.011, respectively.**

***Index Terms*— sEMG, deep learning, long short-term memory network, auto-encoder, domain shift quantification.**

I. Introduction

In HUMAN-MACHINE interfaces (HMI), such as intelligent prostheses or rehabilitation robotics, the identification and prediction of upper-limb motions from surface electromyography (sEMG) signals have received considerable attention [1], [2]. This is largely due to the non-invasiveness of sEMG and the rich neural information that is highly correlated with skeletal muscle contraction [3]. Currently, the machine learning (ML) method has become a prevalent scheme in myoelectric systems. It mainly consists of classification-based gesture recognition and regression-based kinematics/kinetics estimation. The former focuses on the identification of discrete upper-limb activations, enabling dexterous control of multiple degrees of freedom (DoF) [4]. The latter, in contrast, focuses on continuous joint motions to support simultaneous and proportionate control. It has attracted significant interest due to its compatibility with human body movements naturally [5].

To further improve the estimation accuracy and robustness, deep learning (DL) approaches, especially convolutional neural network (CNN), have drawn more and more attentions in both classification [6], [7], [8], [9] and regression [10], [11], [12]. Different from traditional ML models that rely on the extraction and selection of hand-crafted features, CNN-based feature learning requires less expertise in domain knowledge and maintains more information for accurate inference. Nevertheless, a main limitation of CNN is that it inherently focuses on the spatial correlations but is unable to well capture temporal dependencies, whereas upper-limb movements are continuous and preferable to be viewed as time-series data [13]. By contrast, recurrent neural networks (RNNs), particularly long-short term memory network (LSTM), shows great

Manuscript received 31 October 2022; revised 19 April 2023; accepted 26 May 2023. Date of publication 30 May 2023; date of current version 9 June 2023. This work was supported in part by the Shandong Provincial Natural Science Foundation under Grant ZR2020KF012, in part by the Engineering and Physical Sciences Research Council (EPSRC) under Grant EP/S019219/1, and in part by the School of Electronic and Electrical Engineering, University of Leeds. *(Corresponding authors: Tianzhe Bao; Ping Zhou.)*

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the MaPS and Engineering Joint Faculty Research Ethics Committee of University of Leeds, U.K., under Reference No. MEEC 18-006.

Tianzhe Bao is with the School of Rehabilitation Sciences and Engineering, University of Health and Rehabilitation Sciences, Qingdao 266071, China, and also with the School of Electronic and Electrical Engineering, Institute of Robotics, Autonomous System and Sensing, University of Leeds, LS2 9JT Leeds, U.K. (e-mail: tianzhe.bao@uor.edu.cn).

Chao Wang and Zhi-Qiang Zhang are with the School of Electronic and Electrical Engineering, Institute of Robotics, Autonomous System and Sensing, University of Leeds, LS2 9JT Leeds, U.K. (e-mail: elcw@leeds.ac.uk; Z.Zhang3@leeds.ac.uk).

Pengfei Yang is with the School of Computer Science and Technology, Xidian University, Xi'an 710071, China (e-mail: pfyang@xidian.edu.cn).

Sheng Quan Xie is with the Institute of Rehabilitation Engineering, Binzhou Medical University, Yantai 264003, China, and also with the School of Electrical and Electronic Engineering, University of Leeds, LS2 9JT Leeds, U.K. (e-mail: S.Q.Xie@leeds.ac.uk).

Ping Zhou is with the School of Rehabilitation Sciences and Engineering, University of Health and Rehabilitation Sciences, Qingdao 266071, China (e-mail: dr.ping.zhou@outlook.com).

Digital Object Identifier 10.1109/TNSRE.2023.3281455

effectiveness in capturing long-term dependencies by learning contextual information from previous model inputs [14], [15]. To this end, the combination of CNN and RNN/LSTM (such as CNN-RNN, CNN-LSTM, RCNN, etc.) has become one of the-state-of-art frameworks in myoelectric control [13], [16], [17], [18], [19], aiming to better exploit the spacial-temporal correlation of sEMG data via deep feature extraction and sequential regression.

However, it has been noted that the estimation accuracy of pre-trained models tends to decline dramatically in real-world situations, especially in the cross-day scenarios, which severely inhibits the commercialization of myoelectric systems [20], [21], [22]. From the perspective of physiology, it is because the properties of sEMG are non-stable in long-term conditions and can be massively influenced by many external factors such as muscle fatigue, electrode shift, variations of contraction forces, arm position effects, etc. [23], [24], [25], [2], [26]. More specifically, those disturbances result in domain shifts on pre-trained models, as ML/DL methods usually assume that the training and testing data stem from the same underlying distribution [27]. To this end, numerous efforts have been reported, mainly involving dynamic/alternate training [28], [29] to enlarge the labelled dataset and transfer learning [27], [30], [31], [32] to enable parameter adaptation, etc. For instances, Waris et al. [29] investigated the robustness of myoelectric controls in multiple days, and observed that the increase of training size by pulling data from several days tend to improve real-time performance. By contrast, Wang et al. [32] introduced an unsupervised adaptive approach to eliminate the retraining burden over days.

Although the aforementioned studies can help to improve model generalization in some specific scenarios, there still lacks a generic method to predict domain shift impacts on ML/DL models in myoelectric control, especially in the cross-day application. In fact, different from many specific disturbances such as electrode shift, arm position effect, and muscle fatigue, it is less feasible to detect the long-term drift of sEMG via sensors (for example, IMU or computer vision can be used to measure the variation of arm posture or electrode position) or by analysing statistic features of sEMG (such as the mean or median spectrum of sEMG). Therefore, it is of vital importance but also challenging to evaluate the cross-day reliability, whereas domain shift quantification becomes a practical approach.

Recall that variations of sEMG are hard to describe and represent in advance, it is preferable to regard this process as unsupervised anomaly detection [21]. A potential solution is the to use auto-encoders (AE), an unsupervised neural network, to reconstruct model inputs. In this context, the reconstruction errors (RErrors) can work as anomaly scores/indexes [33]. The basic idea behind this approach is that, when AE is trained exclusively with anticipated data, it is supposed to be unfamiliar with unforeseen patterns of anomaly sEMG samples. In another word, AE replicates foreseen sEMG at the output layer with a low reconstruction error but provide a high error rate for significantly varied data. Based on this assumption, an investigation was reported in [34] to reject unknown hand gestures using CNN features of sEMG. However, AE is a non-recurrent structure and thereby may not be well qualified to exploit the temporal dependencies in time-series data, particularly in regression tasks.

In this paper, a reconstruction-based method based on the pairing of LSTM with AE, or LSTM-AE, is introduced for domain shift quantification of CNN-LSTM frameworks. The main purpose is to enhance model usability of DL in cross-day upper-limb motion estimation. To be specific, LSTM-AE reconstructs CNN features of given sEMG data, and the obtained RErrors are utilized to quantify the domain shift that can impact the final estimation performances of CNN-LSTM. Based on the quantification results, threshold strategies can be applied to identify out-of-domain samples. To validate the effectiveness of LSTM-AE, experiments were conducted in both hand gesture recognition and wrist kinematics estimation. The former utilizes a benchmark composed data collected in multi-day classification, and the latter recruits five healthy participants to perform wrist flexion-extension in a two-day experiment. Our main contributions can be summarized as: 1) To the best of our knowledge, this is the first study to investigate domain shift quantification of DL models in cross-day utilization; 2) For a comprehensive research, both classification and regression schemes were involved in experiments; 3) Feature visualization was conducted to provide deeper insights on domain shift impacts.

The reminder of this paper is organized as follows: Section II introduces the framework of domain shift quantification. In this section, a CNN-LSTM hybrid model is firstly presented for upper-limb motion estimation (including both gesture classification and kinematics regression), and the design of LSTM-AE for sequential feature reconstruction is elaborated to quantify domain shift impacts. Section III introduces experiment setups in cross-day classification/regression. Section IV demonstrates the experiment results. Discussions are presented in Section V, and the conclusion is drawn in Section VI.

II. Methodology

Fig. 1 demonstrates the flowchart of proposed method for domain shift quantification in upper-limb motion estimation based on CNN-LSTM model. Firstly, raw sEMG data are converted into matrices for CNN-based feature learning. Then, extracted deep features vectors are further processed to obtain feature sequences for LSTM-based classification or regression. Meanwhile, LSTM-AE is applied to reconstruct feature sequences to quantify domain shift impacts on CNN-LSTM estimation in cross-day utilization. In the following part, we will introduce each module elaborately.

A. CNN-LSTM for Upper-Limb Motion Estimation

In general, there are two types of frameworks in the design of CNN-LSTM, i.e. an end-to-end network composed of both CNN and LSTM layers or a hybrid one cascaded of CNN and LSTM nets separately. In this study, the later is chosen because the extracted CNN features will also be fed into LSTM-AE for domain shift quantification.

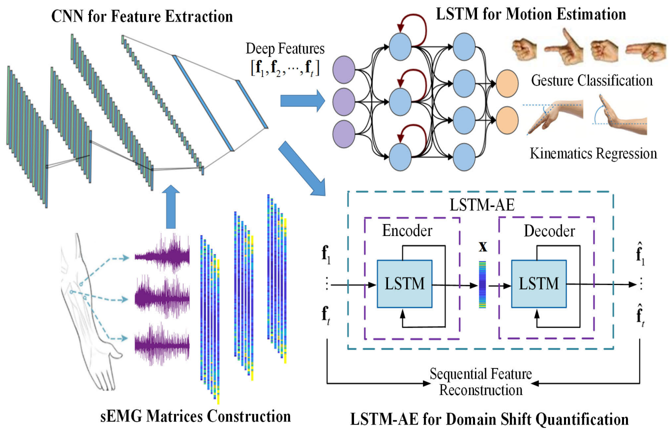

CNN for Feature Extraction
Deep Features
$[\mathbf{f}_1, \mathbf{f}_2, \cdots, \mathbf{f}_t]$
LSTM for Motion Estimation
Gesture Classification
Kinematics Regression
LSTM-AE
Encoder
Decoder
LSTM
x
LSTM
$\mathbf{f}_1$
$\mathbf{f}_t$
$\hat{\mathbf{f}}_1$
$\hat{\mathbf{f}}_t$
Sequential Feature Reconstruction
sEMG Matrices Construction
LSTM-AE for Domain Shift Quantification

Fig. 1. Flowchart of domain shift quantification for CNN-LSTM hybrid model in upper-limb motion estimation using sEMG. It mainly consists of a CNN module for deep feature extraction, a LSTM module for motion estimation, and a LSTM-AE module to quantify domain shift impacts based on feature reconstruction.

1) Deep Feature Extraction: Since CNN is originally designed for processing data in the form of multiple arrays such as images, sEMG matrices/images should be constructed from multi-channels signals. Herein, the one-dimensional (1D) multichannel format [35] is utilized to reduce computational burden. The sEMG data are firstly shaped into $1 \times L \times C$ sized matrices using sliding windows, where L is the length of data in one segment/window and C is the number of sensor channels. Following our previous study [13], the fast Fourier transform (FFT) is conducted to obtain the spectrum of sEMG in each channel, as sEMG matrices are observed to be more discriminative in frequency-domain. More specifically, a L-points frequency vector is obtained in each channel after FFT, where each element of the vector denotes the amplitude of corresponding frequency ranges. Empirically, the double-sided amplitude spectra are maintained such that the shape of sEMG matrices are the same with those in time-domain.

Without loss of generality, basic structures of both classification and regression models are based on LeNet-5. There are 4 convolutional blocks (Conv Block) and 2 fully connected blocks (FC Block). Each Conv Block is composed of a 1D convolutional layer, a batch normalization layer, a leaky ReLU layer, a max-pooling layer, and a dropout layer. The kernel size of convolutional layer is 3, with a boundary padding of 1 and the stride of 1. There are 16 kernels in the 1st and 2nd Conv Block, whilst 32 in the 3rd and 4th block. In each FC Block, the batch normalization layer, leaky ReLU layer and dropout layer are also attached. There are 100 hidden units in the 1st FC Block and 20 in the 2nd. Outputs of the 2nd FC Block work as CNN features of sEMG matrices.

2) Sequential Estimation: LSTM captures the non-linear mappings between CNN features and target gestures or kinematics via a recurrent structure. It contains cycles that feed the network activations from a previous time-step to influence predictions at the current time-step. To enable recurrent learning, successive CNN feature vectors $\mathbf{f}$ are rearranged into feature sequences such as $[\mathbf{f}_1, \cdots, \mathbf{f}_t]$, where t denotes the number of time steps. In this way the historical information can be passed recursively in the whole loop of LSTM. Mathematical descriptions of a LSTM module are as follows.

$$\begin{aligned}
\boldsymbol{i}_i &= \delta\left(\boldsymbol{W}_i\left[\boldsymbol{s}_{i-1}, \mathbf{f}_i\right] + \boldsymbol{b}_i\right)\\
\boldsymbol{m}_i &= \delta\left(\boldsymbol{W}_m\left[\boldsymbol{s}_{i-1}, \mathbf{f}_i\right] + \boldsymbol{b}_m\right)\\
\boldsymbol{o}_i &= \delta\left(\boldsymbol{W}_o\left[\boldsymbol{s}_{i-1}, \mathbf{f}_i\right] + \boldsymbol{b}_o\right)\\
\boldsymbol{c}_i &= \boldsymbol{i}_i \odot \tanh\left(\boldsymbol{W}_c\left[\boldsymbol{s}_{i-1}, \mathbf{f}_i\right] + \boldsymbol{b}_c\right) + \boldsymbol{m}_i \odot \boldsymbol{c}_{i-1}\\
\boldsymbol{s}_i &= \boldsymbol{o}_i \odot \tanh\left(\boldsymbol{c}_i\right)\\
\boldsymbol{y}_i &= \boldsymbol{W}_y \boldsymbol{s}_i + \boldsymbol{b}_y
\end{aligned} \tag{1}$$

where $\boldsymbol{s}_i$ is the hidden state at time-step i in a given feature sequence $[\mathbf{f}_1, \cdots, \mathbf{f}_t]$, $\boldsymbol{c}_i$ is the activation vector, $\boldsymbol{i}_i$ is the input gate, $\boldsymbol{m}_i$ is the forget gate, $\boldsymbol{o}_i$ is the output gate, δ is the logistic sigmoid function, $\boldsymbol{W}$ is the weight matrix in each gate and layer, $\boldsymbol{b}$ is the corresponding bias vector, and $\odot$ is the scalar product.

To reduce over-fitting, only one LSTM layer with 64 hidden units is adopted in both classification and regression, and a dropout layer is attached to improve model generalization. Empirically, a many-to-one structure is applied in LSTM, and the gesture or angle of final time-step works as the label of each sequence sample.

3) Model Training: In this study, a separate training strategy is applied for CNN-LSTM hybrid framework to improve efficiency [13]. Firstly, sEMG matrices, together with gesture labels or wrist angles, are collected to tune CNN. To enable supervised learning, a classification/regression layer is attached to CNN. In the second step, CNN works as a feature extractor. Thereby, parameters of CNN are fixed, and deep feature vectors are extracted from the 2nd fully connected layer. These features are then reconstructed to train LSTM for sequential classification/regression.

In classification, a softmax function $\mathcal{L}_S$ normally works as the training loss in both CNN and LSTM, which can be mathematically expressed as

$$\mathcal{L}_S = -\sum_{k=1}^{N} \log \frac{e^{\boldsymbol{w}_{y_k}^{\mathrm{T}} \boldsymbol{f}_k + \boldsymbol{b}_{y_k}}}{\sum_{j=1}^{m} e^{\boldsymbol{w}_j^{\mathrm{T}} \boldsymbol{f}_k + \boldsymbol{b}_j}} \tag{2}$$

where $\boldsymbol{f}_k$ denotes the kth deep feature that belongs to y_k class, $\boldsymbol{w}$ and $\boldsymbol{b}$ are weights and bias of CNN, respectively. N and m denote the total number of mini-batch samples and the number of classes.

Referring to [34] and [36]. a centre loss $\mathcal{L}_C$ is suggested to train CNN jointly with $\mathcal{L}_S$, which aims to improve the discriminate property of CNN deep features. More specifically, $\mathcal{L}_C$ penalizes distances between deep features and their centre. In this way, instances of the same category tend to cluster more closely around the centre of the category. The mathematical expression of $\mathcal{L}_C$ is

$$\mathcal{L}_C = \sum_{k=1}^{N} \left\| \boldsymbol{f}_k - \boldsymbol{c}_{y_k} \right\|^2 \tag{3}$$

where $\boldsymbol{c}_{y_k}$ is the class centre of deep features that belong to kth class, which is computed by averaging these feature vectors in a mini-batch training set [36].

Fig. 2 demonstrates the design of joint loss function in CNN. As we can see, the joint loss $\mathcal{L}$, which is the sum of $\mathcal{L}_S$

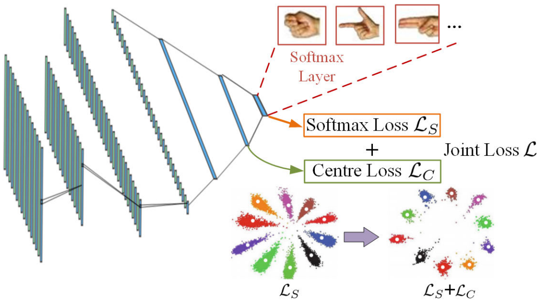

...
Softmax Layer
Softmax Loss $\mathcal{L}_S$
+
Joint Loss $\mathcal{L}$
Centre Loss $\mathcal{L}_C$
$\mathcal{L}_S$
$\mathcal{L}_S$+$\mathcal{L}_C$

Fig. 2. Design of the joint loss to enable a more discriminant supervised learning of CNN feature extractor.

and $\mathcal{L}_C$, maximizes the distance of inter-class instances in the latent space while minimizing that of the intra-class. In this regard, the classification loss of CNN is expressed as

$$\mathcal{L} = \mathcal{L}_S + \lambda\mathcal{L}_C \tag{4}$$

where λ is a hyper-parameter to control the trade-off between two losses.

As for the regression scheme, the mean squared error (MSE) loss is utilized in both CNN and LSTM training. For instance, the training loss of LSTM can be expressed as

$$\mathcal{L} = \sum_{k=1}^{N}\sum_{i=1}^{t}\left\|\boldsymbol{y}_p^{ik} - \boldsymbol{y}_{gt}^{ik}\right\|^2 \tag{5}$$

where $\boldsymbol{y}_p^{ik}$ and $\boldsymbol{y}_{gt}^{ik}$ denote the prediction and ground-truth of each deep feature, respectively. Please recall that this equation is a generic expression for multi-dimensional kinematics regression. A scalar form is used when only one DoF is involved.

It is worth to note that a joint training of CNN and LSTM is also applicable in regression scheme. Nevertheless, a separate training is preferable mainly because this strategy can be more efficient in model training as the input in each time-step of LSTM is a constant vector rather than convolution operations. Besides, the sequential regression part can be easily optimized or replaced without re-training the entire model [37].

B. LSTM-AE for Feature Reconstruction

LSTM-AE is a variation of AE which has achieved great success in feature extraction and internal representation learning. For better clarity, the structure of AE is elaborated first.

AE is a type of deep neural networks trained to reconstruct model input via unsupervised manner. The basic structure of AE is composed of an encoder and a decoder. The encoder part converts its input, such as deep feature $\mathbf{f}_i$ herein, to a latent representation $\mathbf{x}$ through deterministic mapping:

$$\mathbf{x} = \sigma_{en}(\boldsymbol{W}_{en}\boldsymbol{f}_i + \boldsymbol{b}_{en}) \tag{6}$$

where $\sigma()_{en}$ is normally a non-linear function for encoding, $\boldsymbol{W}_{en}$ is the weight matrix and $\boldsymbol{b}_{en}$ is the bias.

By contrast, the decoder is a transformation that maps $\mathbf{x}$ to original input, which can be expressed as:

$$\hat{\mathbf{f}}_i = \sigma_{de}(\boldsymbol{W}_{de}\mathbf{x} + \boldsymbol{b}_{de}) \tag{7}$$

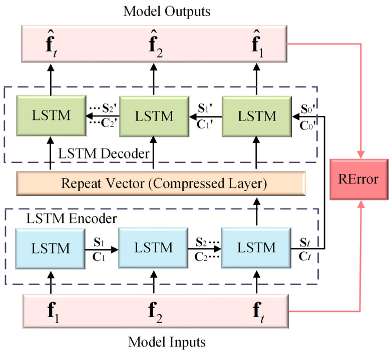


Fig. 3. The unfolded structure of LSTM-AE that is composed of LSTM encoder and LSTM decoder. Please note that a dropout layer is attached to each LSTM layer, and a time distributed dense layer is added to the decoder to obtain final outputs. For the sake of brevity, some of these details are not presented in this figure.

where $\hat{\mathbf{f}}_i$ represents the reconstruction of input $\mathbf{f}_i$, whilst $\sigma()_{de}$, $\boldsymbol{W}_{de}$, and $\boldsymbol{b}_{de}$ denote the mapping function, weight matrix and bias for decoding, respectively.

As illustrated in Fig. 1, LSTM-AE combines the recurrent networks with AE by performing the encoding and decoding process via LSTM networks. More specifically, the encoder LSTM converts the sequence feature $[\mathbf{f}_1, \cdots, \mathbf{f}_t]$ to $\mathbf{x}$, whilst the decoder LSTM reconstructs the output $\left[\hat{\mathbf{f}}_1, \cdots, \hat{\mathbf{f}}_t\right]$ from $\mathbf{x}$ inversely. The unfolded structure of LSTM-AE is shown in Fig. 3. Consistent with CNN-LSTM in classification/regression, one LSTM layer with 64 units is adopted in both the encoder and decoder module of LSTM-AE, and a dropout layer is also attached. Different from the classification/regression task, LSTM-AE uses the many-to-many structure for feature reconstruction.

The principle of training LSTM-AE is to minimize RErrors of given sequential features, such that the reconstructed ones can be close to original inputs. Typically, RError of $[\mathbf{f}_1, \cdots, \mathbf{f}_t]$ is formulated as

$$\text{RError} = \frac{1}{t}\sum_{i=1}^{t}\left\|\mathbf{f}_i - \hat{\mathbf{f}}_i\right\|^2 \tag{8}$$

where the MSE loss is utilized to calculate the difference between input sequence and the reconstructed. For model training, RErrors of a mini-batch of feature sequences will be averaged, and optimal weights of LSTM encoder and decoder can be learned through error backpropagation.

As illustrated in Fig. 4, the implementation of our proposed method mainly includes three steps. Firstly, a CNN-LSTM is trained for motion estimation using labelled sEMG data, where CNN works to extract deep feature vectors from sEMG and LSTM conducts sequential classification or regression. Secondly, a LSTM-AE is trained to reconstruct CNN features via unsupervised learning (following Eq. 6). In the third step, RErrors of given new data are obtained to quantify domain shift impacts that CNN-LSTM suffers. There are several feasible application of LSTM-AE in practice. Typically, threshold strategies can be applied for identification of out-of-domain samples or determining the recalibration of CNN-LSTM in motion estimation.

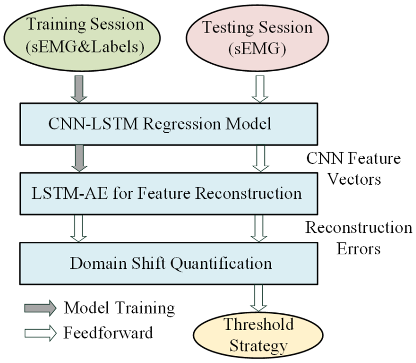

Training Session (sEMG&Labels)
Testing Session (sEMG)
CNN-LSTM Regression Model
CNN Feature Vectors
LSTM-AE for Feature Reconstruction
Reconstruction Errors
Domain Shift Quantification
Model Training
Feedforward
Threshold Strategy

Fig. 4. The implementation protocol of LSTM-AE for domain shift quantification in CNN-LSTM based upper-limb motion estimation.

For brevity, we mainly discuss one of the practical applications of domain shift quantification. More specifically, LSTM-AE is utilized to detect of out-of-domain sEMG samples from normal ones. Recall that, in this study, data in Tr and Ts_1 can be treated as normal ones, whereas those in Ts_2 are regarded as out-of-domain data. Referring to Fig. 4, a threshold strategy is introduced as follows:

$$\text{Given sEMG Data} = \begin{cases} \text{Out-of-domain} & \text{RError} \geq \alpha \\ \text{Normal} & \text{otherwise} \end{cases} \tag{9}$$

where the threshold α can be determined based on the distribution of RErrors obtained in Tr or Ts_1. Herein, α is empirically set to be larger than 95% of RErrors in Tr of each subject, such that the majority of within-day samples will not be misidentified. It is noted that α varies among subjects due to the user-specific nature of sEMG.

III. Experiment Setup

In this study, the effectiveness of LSTM-AE is validated in both hand-gesture classification and joint kinematics regression. The setups of two experiments will be firstly introduced. The signal pre-processing methods and hyper-parameter settings will also be presented.

A. Gesture Classification

The purpose of LSTM-AE is to enhance the usability of CNN-LSTM in cross-day scenarios. Therefore, the classification experiment was conducted using a benchmark dataset, i.e. Non-Invasive Adaptive Hand Prosthetics Database 6 (NinaPro DB6) which was established to investigate the repeatability of sEMG-based hand gesture recognition in cross-day application [20]. The data were collected from 10 healthy participants (noted as P1-P10 for brevity), consisting of 2 sessions (morning and afternoon) each day and lasting for 5 days. As shown in Fig. 5, each session involved 12 repetitions of 7 typical hand grasps in Activities of Daily Living (ADLs). Each grasp repetition lasted approximately 4s, followed by 4s of rest to avoid muscle fatigue. The sEMG signals were measured with 14 Delsys Trigno sEMG Wireless electrodes attached on the forearm. The sampling rate was 2 kHz.

In our experiment, the data in NinaPro DB6 were segmented into three parts, including Tr for model training and Ts_1/Ts_2 for testing. According to experiment results reported in [20], domain shift impacts between sessions are already significant. Therefore, Tr and Ts_1 came from the morning session in the first day (Day 1) to eliminate the cross-day influence. In particular, the first 6 repetitions of each grasp were used for the construction of Tr, and the rest repetitions were used for Ts_1. To validate the cross-day performances, Ts_2 came from the afternoon session in Day 5. In the following part, Ts_1 and Ts_2 are also mentioned as within-day testing set and between-day set, respectively.

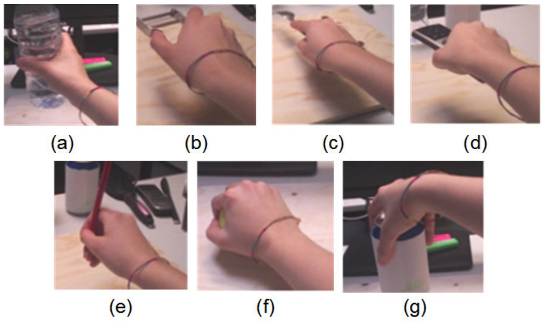


Fig. 5. Classified gestures in NinaPro DB6 [20]. (a) Large Diameter; (b) Adducted Thumb; (c) Index Finger Extension; (d) Medium Wrap; (e) Writing Tripod; (f) Power Sphere; (g) Precision Sphere.

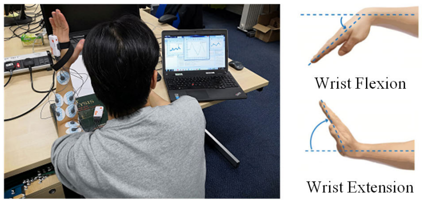


Fig. 6. Data acquisition in the wrist flexion-extension experiment.

B. Kinematics Regression

Due to the lack of public benchmark for cross-day joint kinematics estimation using sEMG, experiments were conducted to obtain our customized dataset. With the approval of MaPS and Engineering joint Faculty Research Ethics Committee of University of Leeds, UK (reference MEEC 18-006), five able-bodied subjects (4 male and 1 female, aged 28-55, noted as S1-S5) were recruited. All subjects were offered the informed and written consent. As shown in Fig. 6, participants were asked to perform smooth and periodical wrist rotations (flexion-extension) lasting about 3 minutes in each trail. Consistent with gesture classification, three trials were collected, i.e. Tr for model training and Ts_1/Ts_2 for testing. In particular, Tr and Ts_1 were conducted subsequently in case of significant domain shift, whilst Ts_2 was obtained about one day later as the cross-day application.

During experiments, subjects seated comfortably in a chair with their forearm limbs relaxed on the table. Twelve bipolar electrodes were placed on the proximal portion of a forearm to collect six channels of sEMG signals using the Shimmer sensing system. Reference electrodes were placed near the wrist. Positions of all electrodes were marked in case of a substantial electrode shift in the second day test. An inertial measurement unit (IMU, composed of an accelerometer, gyroscope, and magnetometer) was attached on the back of hand to calculate the roll, pitch, and yaw angles as the ground-truth of wrist movements. Sampling rates for IMU and sEMG were set as 100 Hz and 1024 Hz, respectively. The data of two sensing systems were then synchronized for model training/testing.

C. Signal Pre-Processing

The sEMG signals in two datasets were both processed using a 3rd order Butterworth high pass filter (20 Hz) and a low pass filter (450 Hz) to remove unusable noise. A Min-Max scaling was applied to normalize sEMG in each channel [38]. For data segmentation, the sliding windows were set to be 100ms with 50ms increment. This is to ensure that sufficient samples are provided for model training in one participant. To be specific, the number of data segments in Tr is about 5000∼6000 in classification and 3000∼4000 in regression. As for data labelling, angles calculated by IMU in each sliding window were averaged to obtain ground-truth labels of corresponding sEMG signals in kinematics estimation. After CNN-based feature extraction, the time duration of reconstructed feature sequence was set to be around 1 second.

D. Hyper-Parameter Setting

In this study, hyper-parameters of CNN-LSTM were mainly identified referring to our previous work [13]. In addition, according to [34], the coefficient λ was set as 0.0005 to balance the softmax loss and centre loss of CNN training in classification schemes. As for LSTM-AE, a grid search based on 5-fold cross-validation was used to tune hyper-parameters using Tr of each participant. Empirically, we report a generic setting which trained model in a 32 sized mini-batch for 100 epochs by stochastic gradient descent (SGD) with a 0.01 learning rate, and the forgetting rate of dropout layer was 0.3.

E. Baseline Methods

Several baseline methods widely used in anomaly detection, including one class support vector machine (OCSVM) [39], Isolation Forests (iForest) [40], and conventional AE [34], were applied for comparison. In particular, OCSVM and AE have been investigated in deep learning applications [34], [39]. As for hyper-parameter setting, a RBF kernel function was applied in OCSVM for non-linear mapping. Other hyper-parameters of OCSVM and iForset were tuned based on data in Tr such that approximately 95% of training samples can be identified as normal ones. Consistent with the structure of LSTM-AE, AE is composed of a three fully-connected layers, with 128, 32 and 128 hidden units in each layer.

F. Evaluation Metric

Since the out-of-domain sEMG data are expected to be detected, the detection performances were evaluated. Herein, an metric named as *Efficiency* was presented. More specifically, $\mathit{Efficiency} = \mathit{Accuracy} - \mathit{ErrorRate}$, where *Accuracy* and *ErrorRate* denotes the detection rates of out-of-domain data in Ts_2 and Ts_1, respectively. It is interesting to find that, if the novelty detection is regarded as a binary classification, *Efficiency* is equivalent to the Youden Index metric [41].

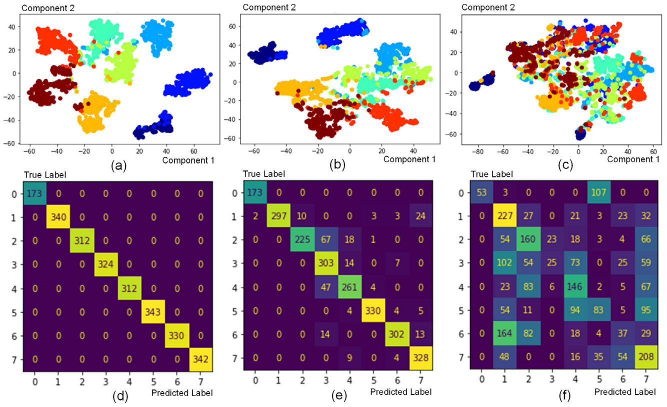


Fig. 7. Distributions of projected CNN features and the associated classification results of CNN-LSTM in Tr, Ts_1, and Ts_2 of a representative subject in Ninapro DB6. (a)-(c) illustrate feature distributions in the latent space, where scatters belonging to the same gesture (including the 'rest' class) are in the same colour. (d)-(f) present the classification performances, where the definitions of 7 gestures can be found in Fig. 5. Label 0 denotes the resting state of the hand.

IV. Results

A. Domain Shift Impacts on CNN-LSTM

As aforementioned, the main function of CNN is to work as feature extractor of sEMG signals. Thereby, it is supposed that domain shift impacts should be well reflected on CNN features in the latent space. For verification, Fig. 7 compared the distributions of extracted CNN features of three trials (Tr, Ts_1, and Ts_2) in gesture classification, associated with the confusion matrices in each trial. High-dimensional CNN features are projected into two dimensions using t-distributed stochastic neighbour embedding (t-SNE) [42] for visualization. It is noted that the feature distribution in Tr, which does not suffer from domain shift impacts, works as the reference.

From Fig. 7(a)-(b) it can be observed that, without too much influence of domain shift impacts on Tr and Ts_1, scatters of the same class are gathering together in the latent space, whilst those belonging to different classes are distributed distantly. To summarize, in these two cases, distributions of CNN features are quite discriminative. On the contrary, scatters in Fig. 7(c), i.e. Ts_2, are heavily overlapped, resulting in a much worse classification result when compared with Tr and Ts_1 (see confusion matrices in Fig. 7(d)-(f)).

According to Fig. 8, similar results can also be obtained in kinematics estimation. As we can see in Fig. 8(a)-(d), scatters with similar angles (i.e. those plotted in similar colour) are clustering closely, whereas those with different angles are distant to each other. A clear and gradual transition from flexion scatters (dark red) to extension scatters (dark blue) can be observed. Accordingly, the estimated trajectory matches the ground-truth accurately. By contrast, in Fig. 8(e) the feature distribution in Ts_2 differs substantially from those of first two datasets. Consequently, the estimated trajectory fails to match the ground-truth (see Fig. 8(f)).

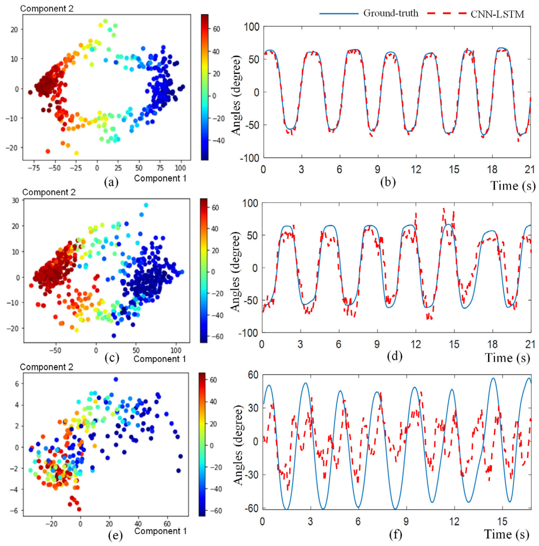


Fig. 8. Distributions of projected CNN features and the associated regression results of CNN-LSTM in Tr, Ts_1, and Ts_2 of a representative subject in our customized experiment. Subfigure (a), (c), and (e) illustrate feature distributions in three datasets, respectively. The angles of scatters (samples) are reflected in colormap, with the dark red representing the positive maximal values in one DoF and dark blue for the negative maximum. Subfigure (b), (d), and (f) present the regression performances accordingly.

B. Domain Shift Quantification via LSTM-AE

Referring to visualization analysis above, domain shift impacts on classification/regression performances are correlated with CNN features of sEMG. Herein, LSTM-AE is utilized to reconstruct CNN features, where data in Ts_2 are considered to be unforeseen due to domain shift in cross-day application. In this regard, RErrors of LSTM-AE should be useful for differentiating CNN features extracted from Ts_1 and Ts_2. For verification, Fig. 9(a) reports the changing of average RErrors (AREs) of regression samples in three datasets during model training. As we can see, AREs of Ts_2 decreases much more slowly than those of Tr and Ts_1, and a substantial difference of AREs between Ts_1 and Ts_2 is observed after convergence.

Histograms of RErrors in Tr, Ts_1 and Ts_2 are illustrated in Fig. 9(b). It can be observed that, the distribution of RErrors in Ts_1, although a little bit skewed, is still close to that in Tr. However, due to domain shift impacts that result in a large number of unforeseen data for LSTM-AE, the majority of RErrors in Ts_2 are much larger than those of other two datasets, thereby the distribution becomes quite separable. The above results are consistent with observations in Fig. 7 and Fig. 8, indicating that domain shift on CNN features will enlarge RErrors of LSTM-AE.

Moreover, Fig. 10 presents the comprehensive analysis on estimation results of CNN-LSTM and reconstruction performances of LSTM-AE in both gesture classification and kinematics estimation. In particular, Fig. 10(a) demonstrates the classification accuracy (ACC) that CNN-LSTM achieved in Tr, Ts_1 and Ts_2 in all participants of gesture classification, and Fig. 10(b) reports the AREs of LSTM-AE accordingly. Consistently, Fig. 10(c)-(d) demonstrate the experiment results

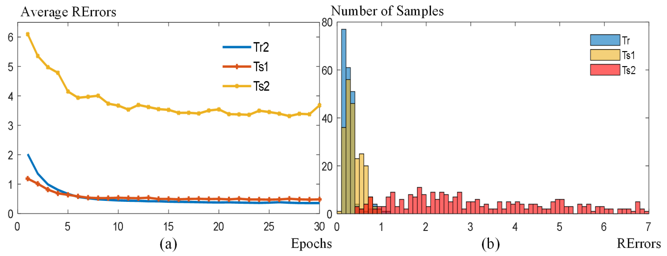


Fig. 9. Reconstruction results of LSTM-AE in Tr, Ts_1 and Ts_2 of a representative subject in kinematics regression: (a) changing of average RErrors (AREs) during model training in first 30 epochs; (b) histograms of RErrors after convergence. The bin width in (b) is set to be 1.

in kinematics estimation. Herein, coefficient of determination [43] (R^2) is utilized to evaluate the regression performance. It is worth to note that R^2 at perfect estimation is equal to one, whilst a negative value means that estimation errors are larger than the variance of target values.

As we can see in Fig. 10(a) and Fig. 10(c), the classification/ regression performances of CNN-LSTM can be well maintained in Ts_1 of most subjects, but drop significantly in Ts_2 due to the domain shift. In accordance, Fig. 10(b) and Fig. 10(d) illustrate that the AREs of LSTM-AE, although also change slightly in Ts_1, increase much more dramatically in Ts_2.

Herein, the Pearson correlation coefficients (CC) was also calculated between ACC/R^2 and AREs of all subjects, and found that the average CC for gesture classification and kinematics estimation is -0.986 ± 0.014 and -0.992 ± 0.011, respectively. From above results it can be concluded that 1) for both gesture classification and kinematics estimation, CNN-LSTM tends to perform poorly in the cross-day utilization; 2) RErrors of between-day testing sets become much larger and also separable from those obtained in within-day sets. In this regard, it is indicated that the LSTM-AE based reconstruction method can work effectively to quantify the domain shift that influences performances of CNN-LSTM.

C. Out-of-Domain Data Detection

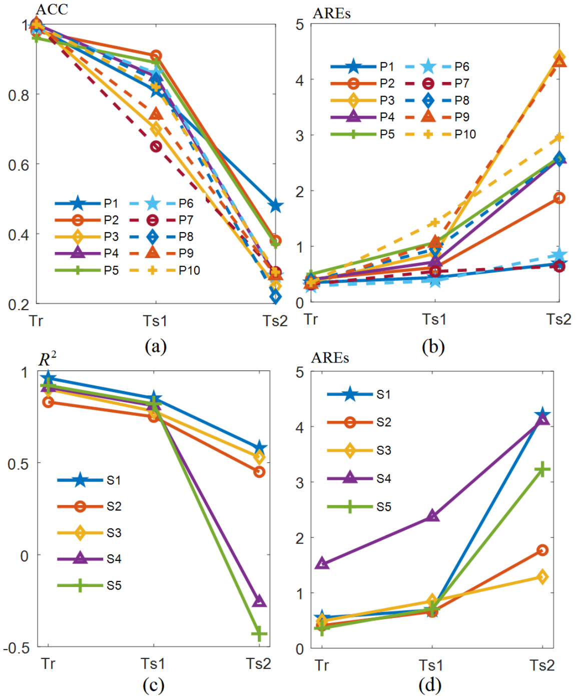
Table I reports the detection rates of out-of-domain data in Ts_1 and Ts_2 in gesture classification, and Table II reports related results in kinematics regression. Please recall that the detection rate obtained in Ts_1 denotes the misidentification of normal data. By contrast, in Ts_2 it is regarded as the detection accuracy. From two tables it is observed that, with acceptable misidentification rates in Ts_1, a large number of between-day samples can be effectively detected using LSTM-AE.

Fig. 11 compares the detection performances of four methods. As we can see, LSTM-AE outperforms the baseline methods in most participants of both gesture classification and kinematics regression. A possible explanation is that, since LSTM-AE works on sequential CNN features whereas other methods only cope with non-sequential ones, distributions of RErrors can be more separable due to the exploitation of temporal dependencies of successive sEMG samples.

TABLE I

DETECTION RATES OF WITHIN-DAY AND BETWEEN-DAY DATA VIA THE THRESHOLD STRATEGY IN GESTURE CLASSIFICATION. IT IS NOTED THAT THE THRESHOLD α IS USER-SPECIFIC AND WAS TO BE LARGER THAN 95% OF RERRORS OF Tr IN EACH PARTICIPANT

Testing Set	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
Within-day (Ts_1)	0.19	0.18	0.17	0.07	0.12	0.10	0.10	0.21	0.29	0.18
Between-day (Ts_2)	0.40	0.77	0.75	0.79	0.51	0.52	0.41	0.99	0.93	0.91



(a) (b) (c) (d)

Fig. 10. Estimation results and reconstruction performances in gesture classification and kinematics regression. (a) Classification accuracy (ACC) of CNN-LSTM in Tr, Ts_1 and Ts_2 of all participants (P1-P10) in gesture classification; (b) Average RErrors (AREs) of LSTM-AE in according experiments; (c) regression performances (coefficient of determination, R^2) of CNN-LSTM in Tr, Ts_1 and Ts_2 of all subjects (S1-S5) in kinematics estimation; (d) Average reconstruction errors (AREs) of LSTM-AE in S1-S5. For better clarity, P1 denotes the first participant in gesture classification and S1 is used for kinematics regression.

TABLE II

DETECTION RATES OF WITHIN-DAY AND BETWEEN-DAY SAMPLES VIA THE THRESHOLD STRATEGY. IT IS NOTED THAT THE THRESHOLD α IS USER-SPECIFIC AND WAS TO BE LARGER THAN 95% OF RERRORS Tr ERRORS IN EACH SUBJECT

Testing Set	S1	S2	S3	S4	S5
Within-day (Ts_1)	0.16	0.09	0.06	0.11	0.23
Between-day (Ts_2)	0.87	0.66	0.71	0.58	0.98

Interestingly, it can also be observed that the detection efficiency varies substantially among participants. For instance, the average *Efficiency* of all methods in P1 (around 0.2) is much lower than that in P8 (more than 0.7). This is because, as reported in Fig. 10, the estimation performances of CNN-LSTM were much less impacted in the between-day testing set of P1 (classification accuracy maintains at 0.48 in Ts_2). By contrast, the accuracy drops substantially (from 0.84 to 0.22) in datasets of P8.

TABLE III

AUC OF ALL PARTICIPANTS IN CLASSIFICATION TASK

Participant	LSTM-AE	AE	OCSVM	iForest
P1	0.66	0.62	0.58	0.54
P2	0.88	0.79	0.61	0.73
P3	0.90	0.88	0.91	0.91
P4	0.91	0.86	0.88	0.87
P5	0.83	0.79	0.53	0.59
P6	0.85	0.80	0.66	0.74
P7	0.73	0.69	0.61	0.61
P8	0.94	0.93	0.96	0.95
P9	0.91	0.86	0.84	0.85
P10	0.92	0.93	0.90	0.76

TABLE IV

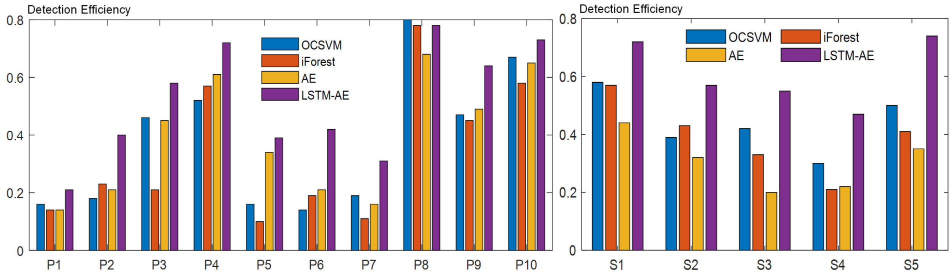
AUC OF ALL SUBJECTS IN REGRESSION TASK

Subject	LSTM-AE	AE	OCSVM	iForest
S1	0.92	0.81	0.87	0.82
S2	0.88	0.75	0.81	0.78
S3	0.89	0.66	0.81	0.62
S4	0.74	0.53	0.59	0.56
S5	0.94	0.81	0.84	0.79

V. DISCUSSION

As revealed in previous literature [20], [21], [22], [30], [31], [32], properties of sEMG is non-stable and time-varying, resulting in obstacles that hinder accurate and reliable motion estimation in real-world myoelectric control. Therefore, an effective evaluation on model reliability is of vital importance. In this study, a novel method is presented for domain shift quantification of a prevalent CNN-LSTM hybrid framework in both gesture classification and kinematics regression. Compared with many other DL methods, this framework is preferred because 1) it achieves state-of-the-art performances by exploiting the inherent spatial-temporal dependencies among sEMG signals; 2) in this two-step structure, CNN features can be further utilized by LSTM-AE in a very feasible way.

From experiment results it can be observed that the quantification performances, i.e. RErrors of LSTM-AE, are highly correlated with distribution of CNN features and also with the classification/regression accuracies of CNN-LSTM. Therefore, it is inferred that domain shift impacts on LSTM estimator can be well reflected by the distribution of CNN features in latent space, and these impacts can be further quantified by reconstruction errors of CNN features using LSTM-AE. In fact, the domain shift impacts on CNN feature has also been utilized in metric learning, where the alignment of CNN features in the latent space across datasets (i.e. the source domain and target domain) can help to improve the performance of CNN in the cross-day scenarios.



(a) Gesture Classification (b) Kinematics Estimation

Fig. 11. Detection efficiency of OCSVM, iForest, AE, and LSTM-AE in gesture classification and kinematics estimation.

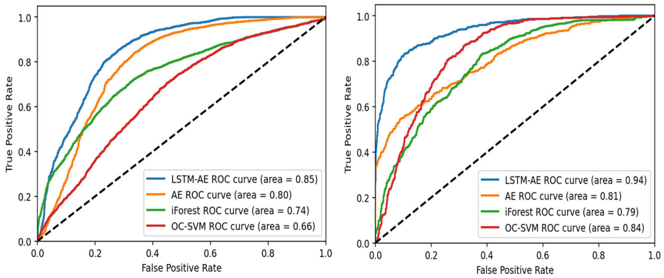
TABLE V
Classification Results of Conventional LSTM and Reconstruction Errors Using LSTM-AE

Results/Testing Sets		P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
ACC	Ts_1	0.81	0.85	0.72	0.79	0.84	0.82	0.67	0.82	0.71	0.78
	Ts_2	0.39	0.28	0.27	0.20	0.33	0.20	0.30	0.24	0.26	0.30
AREs	Ts_1	0.29	0.49	0.77	0.94	1.48	0.45	0.60	0.88	0.88	2.02
	Ts_2	0.56	1.17	3.90	3.03	2.84	1.16	1.01	2.13	3.51	3.66

To verify performances of LSTM-AE in cross-day gesture classification, the NinaPro DB6 is used as a benchmark in this work. NinaPro DB6 is composed of 12 inter-session datasets across 5 days, and the contributor, Palermo et al. [20], reached an inter-session accuracy of 25.4% based on a Random Forest classifier. This is in general similar to results reported in Fig. 10(a) (see ACC of participants in Ts_2), but is much lower than those reached on other datasets with a similar number of classes and sensors. The main reason is that the hand movements of NinaPro DB6, as shown in Fig. 5, are all grasps that are much less diverse and discernable than the hand gestures in ordinary datasets. As for kinematics estimation, the within-day performances of CNN-LSTM are among 0.8-0.9 in R^2. These results are consistent with our previous study [13]. From Fig. 9 and Fig. 10 it can be observed that RErrors are distinguishable between within-day data and between-day ones. As aforementioned, LSTM-AE is less able to reconstruct unforeseen CNN features well compared to foreseen ones, thereby RErrors of out-of-domain data should be comparatively larger than those of normal ones.

As aforementioned in Section III-F, the detection of out-of-domain sEMG samples, i.e. between-day data, can be regarded as binary classification. Thereby, area under the curve (AUC), which is a metric obtained via computing the receiver operating characteristic (ROC) curve of true positive rate (TPR) and false positive rate (FPR) under different thresholds, was applied. Fig. 12 demonstrates performances of LSTM-AE and baseline methods in representative subjects/participants of gesture classification and kinematics estimation, respectively. Table III and Table IV summarize the AUC results of all methods. As is shown in the figures/tables, LSTM-AE could outperform AE, OCSVM, and iForest in almost all cases. In addition, AUC of different subjects/participants varies substantially. These observations are in general consistent with results evaluated via detection efficiency.

In our study, we mainly focus on the application of LSTM-AE on CNN-LSTM framework. This is motivated by both [34] (combines AE and CNN for outlier detection) and our experiments (Fig. 7 and Fig. 8) which reveal that domain shift impacts could be sufficiently reflected on CNN features in the latent space. Thereby, it could be inferred that LSTM-AE is able to collaborate well with other variations of CNN which substitutes LetNet-5 used in the presented CNN-LSTM framework, where deep features can be extracted from fully connected layers feasibly.



(a) ROC curves in classification (P6) (b) ROC curves in regression (S5)

Fig. 12. ROC curves and AUC results of representative subjects/participants in classification (P6) and regression (S5). ROC curves were computed based on true positive rate (TPR) and false positive rate (FPR) under varied thresholds. It is noted that OVSVM and iForset can also provide scores of sEMG samples.

It is also of our interest to investigate the generalization of LSTM-AE in other deep learning frameworks. Please recall that recurrent models (i.e. RNN, LSTM, GRU, etc.) are preferred since LSTM-AE is proposed for the processing of time-series data. Without loss of generality, LSTM was selected as the backbone for classification and regression. A widely applied temporal-spatial feature set [34] was adopted to extract hand-crafted features of sEMG.

Table V and Table VI summarize the experiment results of LSTM-based classification or regression (ACC/R^2) as well as reconstruction errors (AREs) of LSTM-AE. To analyse more sufficiently, results of Ts_2 were obtained as the average of those in afternoon sessions of Day2-Day5. As we can see,

TABLE VI
REGRESSION RESULTS AND CORRESPONDING RECONSTRUCTION ERRORS

Results/Testing Sets		S1	S2	S3	S4	S5
R^2	Ts_1	0.82	0.74	0.70	0.83	0.78
	Ts_2	0.12	0.19	0.44	-0.06	-0.93
AREs	Ts_1	0.75	0.81	0.80	1.94	0.88
	Ts_2	3.94	2.13	1.41	4.32	3.44

AREs of LSTM-AE increase substantially when classification or regression performances of LSTM decay due to domain shift in cross-day application, which is consistent with results of CNN-LSTM. A possible explanation is that domain shift impacts can also be well reflected on the selected hand-crafted features, indicating potential generalization of LSTM-AE for domain shift quantification in some other deep learning or even machine learning models.

VI. Conclusion

In this paper, a reconstruction-based method using LSTM-AE is presented for the domain shift quantification in the cross-day application of CNN-LSTM in myoelectric control. In our experiments, it can be observed that CNN features of short-term and cross-day dataset are distributed differently in the latent space, which is associated with the degradation of estimation accuracies of LSTM using CNN features. Therefore, the proposed LSTM-AE provides an effective way for domain shift quantification. Furthermore, it can be utilized to detect out-of-domain samples or indicate model re-calibration, helping to improve system reliability in myoelectric control.

Acknowledgment

Author Contributions

Conceptualization: Tianzhe Bao, Zhi-Qiang Zhang, and Ping Zhou; methodology: Tianzhe Bao, and Chao Wang; validation: writing—original draft preparation and writing—review and editing: Tianzhe Bao; visualization: Tianzhe Bao, and Pengfei Yang; supervision: Sheng Quan Xie, Zhi-Qiang Zhang, and Ping Zhou; and funding acquisition: Zhi-Qiang Zhang, and Ping Zhou. All authors have read and agreed to the published version of the manuscript.

References

[1] D. Xiong, D. Zhang, X. Zhao, and Y. Zhao, "Deep learning for EMG-based human-machine interaction: A review," *IEEE/CAA J. Autom. Sinica*, vol. 8, no. 3, pp. 512–533, Mar. 2021.

[2] H. Xu and A. Xiong, "Advances and disturbances in sEMG-based intentions and movements recognition: A review," *IEEE Sensors J.*, vol. 21, no. 12, pp. 13019–13028, Jun. 2021.

[3] V. Medved, S. Medved, and I. Kovač, "Critical appraisal of surface electromyography (sEMG) as a taught subject and clinical tool in medicine and kinesiology," *Frontiers Neurol.*, vol. 11, Oct. 2020, Art. no. 560363.

[4] F. S. Botros, A. Phinyomark, and E. J. Scheme, "Electromyography-based gesture recognition: Is it time to change focus from the forearm to the wrist?" *IEEE Trans. Ind. Informat.*, vol. 18, no. 1, pp. 174–184, Jan. 2022.

[5] A. W. Shehata, E. J. Scheme, and J. W. Sensinger, "Evaluating internal model strength and performance of myoelectric prosthesis control strategies," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 5, pp. 1046–1055, May 2018.

[6] W. Wei, Q. Dai, Y. Wong, Y. Hu, M. Kankanhalli, and W. Geng, "Surface-electromyography-based gesture recognition by multi-view deep learning," *IEEE Trans. Biomed. Eng.*, vol. 66, no. 10, pp. 2964–2973, Oct. 2019.

[7] X. Lv et al., "Gesture recognition based on sEMG using multi-attention mechanism for remote control," *Neural. Comput. Appl.*, vol. 35, pp. 1–11, Jan. 2022.

[8] Y. Lin, R. Palaniappan, P. De Wilde, and L. Li, "Reliability analysis for finger movement recognition with raw electromyographic signal by evidential convolutional networks," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 96–107, 2022.

[9] T. Bao, S. A. R. Zaidi, S. Q. Xie, P. Yang, and Z. Zhang, "CNN confidence estimation for rejection-based hand gesture classification in myoelectric control," *IEEE Trans. Human-Mach. Syst.*, vol. 52, no. 1, pp. 99–109, Feb. 2022.

[10] A. Ameri, M. A. Akhaee, E. Scheme, and K. Englehart, "Regression convolutional neural network for improved simultaneous EMG control," *J. Neural Eng.*, vol. 16, no. 3, Jun. 2019, Art. no. 036015.

[11] W. Yang, D. Yang, Y. Liu, and H. Liu, "Decoding simultaneous multi-DOF wrist movements from raw EMG signals using a convolutional neural network," *IEEE Trans. Human-Mach. Syst.*, vol. 49, no. 5, pp. 411–420, Oct. 2019.

[12] D. Yang and H. Liu, "An EMG-based deep learning approach for multi-DOF wrist movement decoding," *IEEE Trans. Ind. Electron.*, vol. 69, no. 7, pp. 7099–7108, Jul. 2022.

[13] T. Bao, S. A. R. Zaidi, S. Xie, P. Yang, and Z. Zhang, "A CNN-LSTM hybrid model for wrist kinematics estimation using surface electromyography," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–9, 2021.

[14] 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.

[15] R. N. Khushaba, E. Scheme, A. H. Al-Timemy, A. Phinyomark, A. Al-Taee, and A. Al-Jumaily, "A long short-term recurrent spatial–temporal fusion for myoelectric pattern recognition," *Expert Syst. Appl.*, vol. 178, Sep. 2021, Art. no. 114977.

[16] M. Simão, P. Neto, and O. Gibaru, "EMG-based online classification of gestures with recurrent neural networks," *Pattern Recognit. Lett.*, vol. 128, pp. 45–51, Dec. 2019.

[17] R. Antonius and H. Tjahyadi, "Electromyography gesture identification using CNN-RNN neural network for controlling quadcopters," *J. Phys. Conf.*, vol. 1858, no. 1, 2021, Art. no. 012075.

[18] T. Bao, Y. Zhao, S. A. R. Zaidi, S. Xie, P. Yang, and Z. Zhang, "A deep Kalman filter network for hand kinematics estimation using sEMG," *Pattern Recognit. Lett.*, vol. 143, pp. 88–94, Mar. 2021.

[19] N. K. Karnam, S. R. Dubey, A. C. Turlapaty, and B. Gokaraju, "EMGHandNet: A hybrid CNN and bi-LSTM architecture for hand activity classification using surface EMG signals," *Biocybernetics Biomed. Eng.*, vol. 42, no. 1, pp. 325–340, Jan. 2022.

[20] F. Palermo, M. Cognolato, A. Gijsberts, H. Müller, B. Caputo, and M. Atzori, "Repeatability of grasp recognition for robotic hand prosthesis control based on sEMG data," in *Proc. Int. Conf. Rehabil. Robot. (ICORR)*, Jul. 2017, pp. 1154–1159.

[21] L. Wu, X. Zhang, X. Chen, and X. Chen, "Visualized evidences for detecting novelty in myoelectric pattern recognition using 3D convolutional neural networks," in *Proc. 41st Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2019, pp. 2641–2644.

[22] 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.

[23] E. C. Hill et al., "Effect of sex on torque, recovery, EMG, and MMG responses to fatigue," *J. Musculoskelet Neuronal Interact.*, vol. 16, no. 4, p. 310, 2016.

[24] R. Kusche and M. Ryschka, "Combining bioimpedance and EMG measurements for reliable muscle contraction detection," *IEEE Sensors J.*, vol. 19, no. 23, pp. 11687–11696, Dec. 2019.

[25] A. Waris, I. K. Niazi, M. Jamil, K. Englehart, W. Jensen, and E. N. Kamavuako, "Multiday evaluation of techniques for EMG-based classification of hand motions," *IEEE J. Biomed. Health Informat.*, vol. 23, no. 4, pp. 1526–1534, Jul. 2019.

[26] T. Bao, S. Q. Xie, P. Yang, P. Zhou, and Z.-Q. Zhang, "Toward robust, adaptiveand reliable upper-limb motion estimation using machine learning and deep learning—A survey in myoelectric control," *IEEE J. Biomed. Health Informat.*, vol. 26, no. 8, pp. 3822–3835, Aug. 2022.

[27] T. Bao, S. A. R. Zaidi, S. Xie, P. Yang, and Z. Zhang, "Inter-subject domain adaptation for CNN-based wrist kinematics estimation using sEMG," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 1068–1078, 2021.

[28] S. Benatti, F. Montagna, V. Kartsch, A. Rahimi, D. Rossi, and L. Benini, "Online learning and classification of EMG-based gestures on a parallel ultra-low power platform using hyperdimensional computing," *IEEE Trans. Biomed. Circuits Syst.*, vol. 13, no. 3, pp. 516–528, Jun. 2019.

[29] A. Waris, I. Mendez, K. Englehart, W. Jensen, and E. N. Kamavuako, "On the robustness of real-time myoelectric control investigations: A multiday Fitts' law approach," *J. Neural Eng.*, vol. 16, no. 2, Apr. 2019, Art. no. 026003.

[30] A. Ameri, M. A. Akhaee, E. Scheme, and K. Englehart, "A deep transfer learning approach to reducing the effect of electrode shift in EMG pattern recognition-based control," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 28, no. 2, pp. 370–379, Feb. 2020.

[31] X. Sheng, B. Lv, W. Guo, and X. Zhu, "Common spatial–spectral analysis of EMG signals for multiday and multiuser myoelectric interface," *Biomed. Signal Process. Control*, vol. 53, Aug. 2019, Art. no. 101572.

[32] H. Wang, P. Huang, T. Xu, G. Li, and Y. Hu, "Towards zero retraining for multiday motion recognition via a fully unsupervised adaptive approach and fabric myoelectric armband," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 217–225, 2022.

[33] Y. Xia, X. Cao, F. Wen, G. Hua, and J. Sun, "Learning discriminative reconstructions for unsupervised outlier removal," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1511–1519.

[34] L. Wu, X. Zhang, X. Zhang, X. Chen, and X. Chen, "Metric learning for novel motion rejection in high-density myoelectric pattern recognition," *Knowl.-Based Syst.*, vol. 227, Sep. 2021, Art. no. 107165.

[35] Y. Hu, Y. Wong, W. Wei, Y. Du, M. Kankanhalli, and W. Geng, "A novel attention-based hybrid CNN-RNN architecture for sEMG-based gesture recognition," *PLoS ONE*, vol. 13, no. 10, Oct. 2018, Art. no. e0206049.

[36] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 499–515, doi: 10.1007/978-3-319-46478-7_31.

[37] Z. Wu, X. Wang, Y.-G. Jiang, H. Ye, and X. Xue, "Modeling spatial–temporal clues in a hybrid deep learning framework for video classification," in *Proc. 23rd ACM Int. Conf. Multimedia*, Oct. 2015, pp. 461–470.

[38] Z. Huang, C. Yang, X. Zhou, and T. Huang, "A hybrid feature selection method based on binary state transition algorithm and ReliefF," *IEEE J. Biomed. Health Informat.*, vol. 23, no. 5, pp. 1888–1898, Sep. 2019.

[39] S. M. Erfani, S. Rajasegarar, S. Karunasekera, and C. Leckie, "High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning," *Pattern Recognit.*, vol. 58, pp. 121–134, Oct. 2016.

[40] L. F. Tony, T. K. Ming, and Z. Zhi-Hua, "Isolation-based anomaly detection," *ACM Trans. Knowl. Discovery Data*, vol. 6, no. 1, p. 3, 2012.

[41] R. Fluss, D. Faraggi, and B. Reiser, "Estimation of the youden index and its associated cutoff point," *Biometrical J.*, vol. 47, no. 4, pp. 458–472, Aug. 2005.

[42] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

[43] A. d'Avella, A. Portone, L. Fernandez, and F. Lacquaniti, "Control of fast-reaching movements by muscle synergy combinations," *J. Neurosci.*, vol. 26, no. 30, pp. 7791–7810, Jul. 2006.