



UNIVERSITY OF LEEDS

This is a repository copy of *A CNN-LSTM Hybrid Model for Wrist Kinematics Estimation Using Surface Electromyography*.

White Rose Research Online URL for this paper:

<https://eprints.whiterose.ac.uk/167868/>

Version: Accepted Version

---

**Article:**

Bao, T [orcid.org/0000-0002-1103-2660](https://orcid.org/0000-0002-1103-2660), Zaidi, SAR [orcid.org/0000-0003-1969-3727](https://orcid.org/0000-0003-1969-3727), Xie, S [orcid.org/0000-0002-8082-9112](https://orcid.org/0000-0002-8082-9112) et al. (2 more authors) (2020) A CNN-LSTM Hybrid Model for Wrist Kinematics Estimation Using Surface Electromyography. IEEE Transactions on Instrumentation and Measurement, 70. 2503809. ISSN 0018-9456

<https://doi.org/10.1109/tim.2020.3036654>

---

© 2020 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

**Reuse**

Items deposited in White Rose Research Online are protected by copyright, with all rights reserved unless indicated otherwise. They may be downloaded and/or printed for private study, or other acts as permitted by national copyright laws. The publisher or other rights holders may allow further reproduction and re-use of the full text version. This is indicated by the licence information on the White Rose Research Online record for the item.

**Takedown**

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing [eprints@whiterose.ac.uk](mailto:eprints@whiterose.ac.uk) including the URL of the record and the reason for the withdrawal request.



[eprints@whiterose.ac.uk](mailto:eprints@whiterose.ac.uk)  
<https://eprints.whiterose.ac.uk/>

# A CNN-LSTM Hybrid Model for Wrist Kinematics Estimation Using Surface Electromyography

Tianzhe Bao, *Student Member, IEEE*, Syed Ali Raza Zaidi, *Member, IEEE*, Shengquan Xie, *Senior Member, IEEE*, Pengfei Yang, *Member, IEEE*, and Zhi-Qiang Zhang, *Member, IEEE*

**Abstract**—Convolutional neural network (CNN) has been widely exploited for simultaneous and proportional myoelectric control due to its capability of deriving informative, representative and transferable features from surface electromyography (sEMG). However, muscle contractions have strong temporal dependencies but conventional CNN can only exploit spatial correlations. Considering that long short-term memory neural network (LSTM) is able to capture long-term and non-linear dynamics of time-series data, in this paper we propose a CNN-LSTM hybrid model to fully explore the temporal-spatial information in sEMG. Firstly, CNN is utilized to extract deep features from sEMG spectrum, then these features are processed via LSTM-based sequence regression to estimate wrist kinematics. Six healthy participants are recruited for the participatory collection and motion analysis under various experimental setups. Estimation results in both intra-session and inter-session evaluations illustrate that CNN-LSTM significantly outperforms CNN, LSTM and several representative machine learning approaches, particularly when complex wrist movements are activated.

**Index Terms**—sEMG, wrist kinematics estimation, deep learning, convolutional neural network, long short-term memory network, hybrid model.

## I. INTRODUCTION

**D**URING the past decades, there has been considerable attention given to surface electromyography (sEMG) in driving active prosthetic hands [1] and upper limb exoskeleton robotics [2]. To achieve intuitive myoelectric control, machine learning (ML) approaches, i.e. classifier-based pattern recognition (PR) and regression, have been extensively investigated in recent literature. Unlike PR-based methods which discriminate hand gestures in a discrete and sequential manner [3], regression models focus on continuous wrist kinematics estimation [4] and thus can promote simultaneous and proportional control in multiple degrees of freedoms (DoF). Several ML-based regression methods, including linear regression (LR), artificial neural network (ANN), kernel ridge regression, support vector regression (SVR) and random forest (RF), have been extensively exploited in both off-line simulations [5–9]

This work was supported in part by Engineering and Physical Sciences Research Council (EPSRC) (Grant No. EP/S019219/1), School of Electronic and Electrical Engineering, University of Leeds, and the National Natural Science Foundation of China under Grant 61962019 and Grant 61702395. (*Corresponding author: Zhi-Qiang Zhang, Pengfei Yang.*)

Tianzhe Bao, Syed Ali Raza Zaidi, Shengquan Xie and Zhi-Qiang Zhang are with Institute of Robotics, Autonomous System and Sensing, School of Electrical and Electronic Engineering, University of Leeds, LS2 9JT, UK (e-mail: {eltb, s.a.zaidi, s.q.xie, z.zhang3}@leeds.ac.uk).

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

and real-time prosthetic control [1]. However, ML techniques rely deeply on manual feature extraction [10], i.e. feature engineering. Due to the stochastic nature of sEMG signals [11] and serious crosstalk among muscles, useful information can be easily obscured in hand-crafted features.

Deep learning (DL), particularly the convolutional neural network (CNN), is now providing a new perspective for feature learning/extraction via layer-by-layer processing [12, 13]. Promising results have been achieved in sEMG-based hand gesture recognition in the past few years. For instance, Park et al. presented a single stream CNN and evaluated the performance of DL learning via inter-subject estimations [14]. Atzori et al. made a comprehensive comparison between CNN and several ML classifiers based on the NinaPro dataset [15]. Du et al. presented an AdaBN-based deep domain adaptation scheme for inter-session recognition and conducted evaluations with two more public datasets (CSL-HDEMG and CapgMyo) [16]. Wei et al. proposed a two-stage multi-stream CNN to learn the correlations between individual muscles [17]. Ding et al. proposed a parallel multiple-scale convolution architecture which exploited different size of kernel filters [18]. In addition, there are several pilot studies on regression-based wrist kinematics estimation. For instance, Ameri et al. investigated a CNN-based regression technique which outperformed a traditional SVR-based scheme in an online Fitts' law test [19]. Yang et al. presented several data-augmentation approaches for CNN in decoding 3-DoF wrist movements [20], and verified that the proposed CNN structure outperformed SVR significantly when confounding factors were involved [21]. Although CNN is good at extracting spatial correlations of multi-channel sEMG signals, it inherently ignores the temporal information during continuous muscle contractions.

Most recently, many researchers begin to implement the long short-term memory network (LSTM) for sEMG-based hand pose estimation. For example, Quivira et al. applied LSTM to build an accurate regression model for predicting hand joint kinematics from sEMG features [22]. Teban et al. claimed that LSTM performed better than a non-recurrent ANN in replicating a non-linear mechanism of a real human hand [23]. He et al. combined LSTM with ANN to exploit both the dynamic and static information of sEMG [24]. Ali et al. validated that a bidirectional LSTM with attention mechanism could outperform other tested recurrent neural networks (RNN) in sEMG-based hand gesture recognition [25]. Despite that LSTM shows great effectiveness in capturing temporal dependencies based on learning contextual information from past inputs [26], all those pilot studies have only

applied conventional hand-crafted features rather than deep spatial features in their regression process. To this end, The combination of CNN and RNN/LSTM is now becoming a trend in sEMG-based motion estimation. In particular, Xia et al. [27] proposed a recurrent convolutional neural networks (RCNN) architecture to integrate CNN and LSTM layers for the tracking of arm movements. Huang et al. [28] applied a similar architecture in hand gesture classification and verified that the proposed model outperformed SVM in three different exercises of Ninapro Database 2.

Inspired by advantages and limitations of CNN and LSTM, in this paper we propose a CNN-LSTM hybrid model to combine deep feature extraction and sequence regression efficiently, so that the temporal-spatial correlations of sEMG can be fully exploited. With deep features extracted from CNN and then processed by LSTM, wrist kinematics in single/multiple DoFs can be reconstructed accurately. Compared with conventional CNN and LSTM, CNN-LSTM is more robust to localized distortions along time. In this study, six healthy participants take part in experiments to perform a series of wrist movements. Experimental results indicate that CNN-LSTM outperforms CNN, LSTM and several representative ML approaches in both intra-session and inter-session scenarios, especially when complex wrist movements are activated in multi-DoFs. Contributions of this paper can be summarized in three aspects: 1) it firstly investigates the effective combination of CNN and LSTM in wrist kinematics estimation using sEMG signals; 2) a separate training strategy is utilized to improve the computational efficiency and model feasibility; 3) visual explorations of two types of features indicate that distributions of CNN features can be better correlated with wrist motions than many hand-crafted features.

The remainder of this paper is structured as follows. Section II describes the proposed hybrid model, where the implementation of deep feature extraction and sequence regression are separately elaborated. Section III introduces experimental setups and Section IV presents estimation results in both intra-session and inter-session evaluations. In Section V a discussion is presented to analyse model merits, limitations, and future work. The conclusion is then drawn in this Section VI.

## II. CNN-LSTM HYBRID MODEL

As illustrated in Fig. 1, our CNN-LSTM model consists of two steps: the first step is to implement CNN for feature extraction and the second step is to construct LSTM for sequence regression. In the first step CNN is utilised to extract deep feature vector  $\mathbf{f}$  from the sEMG matrix  $\mathbf{X}$  which is constructed on a segment of multi-channel sEMG signals. In the second step, successive deep feature vectors are rearranged into a series of feature sequences, such as  $[\mathbf{f}_1, \mathbf{f}_2 \cdots \mathbf{f}_k]$ ,  $[\mathbf{f}_2, \mathbf{f}_3 \cdots \mathbf{f}_{k+1}]$ , etc. The parameter  $k$  is the number of feature vectors in a feature sequence, which denotes the time-steps in recurrent regression. A LSTM is built to convert  $[\mathbf{f}_1, \mathbf{f}_2 \cdots \mathbf{f}_k]$  into wrist angles  $[y_1, y_2, \cdots y_k]$ . In this study, we adopt the last output  $y_k$  as the final observation of this sequence. In the following part we will elaborate the implementation of CNN and LSTM, together with the training process of each model.

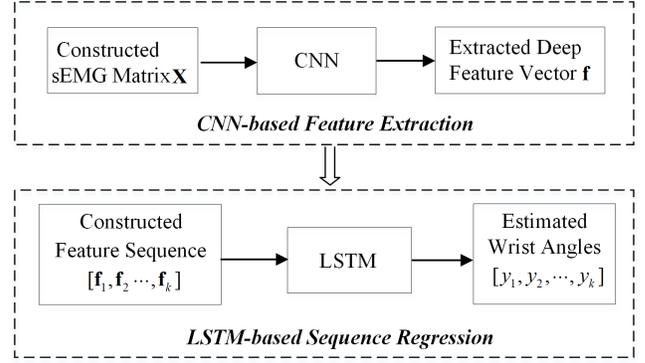


Fig. 1: Block diagram of CNN-LSTM hybrid model.

### A. CNN-based Deep Feature Extraction

1) *Construction of sEMG Matrices:* Firstly, we use the sliding window method to split multi-channel sEMG into segments, and then signals in one segment are rearranged into a  $1 \times L \times N$  matrix [27, 29]. Herein  $L$  corresponds to the length of a sliding window and  $N$  is in accordance with the number of sensor channels. By applying fast Fourier transform (FFT) on each channel, the spectrum-based sEMG matrix can be obtained as CNN inputs.

2) *CNN Architecture:* As illustrated in Fig. 2, the presented CNN consists of 4 convolutional blocks (Conv Block) and 2 fully connected blocks (FC Block). Each Conv Block has a convolutional layer, a batch normalization layer, a leaky ReLU layer, a max-pooling layer and a dropout layer. The convolution layer uses a kernel size of 3, a boundary padding of 1 and the stride of 1. There are 16 kernels in the 1<sup>st</sup> and 2<sup>nd</sup> Conv Block whilst 32 in the 3<sup>rd</sup> and 4<sup>th</sup> block. The batch normalization layer is attached to mitigate alternation made by convolutional layers [30]. As suggested in our previous work [31], the leaky ReLU layer is used in case of the dying ReLU problem [32]. The max-pooling layer (a pool size of 3 and a stride of 1) is added for sub-sampling while a dropout layer is attached for regularization. In each FC Block, the batch normalization layer, leaky ReLU layer and dropout layer are added subsequently to the fully connected layer. There are 100 hidden units in the 1<sup>st</sup> FC Block and 20 in the 2<sup>nd</sup>. Outputs of the 2<sup>nd</sup> FC Block will be utilized as the deep feature  $\mathbf{f}$  for LSTM-based sequence regression. Thus the dimension of CNN features is 20.

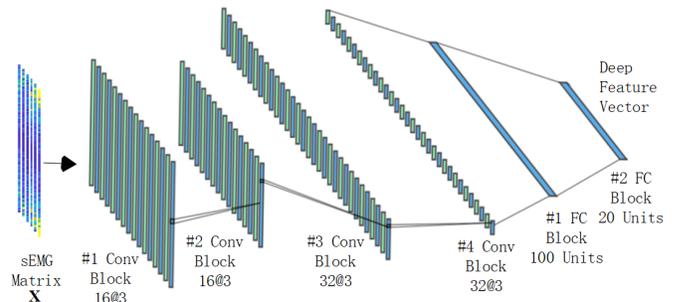


Fig. 2: The single stream CNN architecture for deep feature extraction.

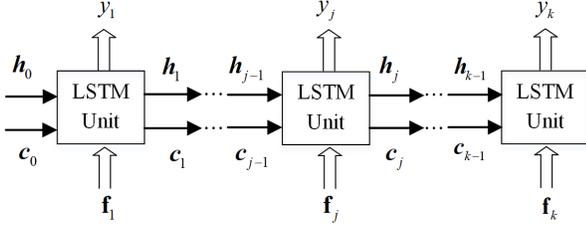


Fig. 3: The unfolded chain structure of LSTM in time sequence with deep CNN features.

### B. LSTM-based Sequence Regression

1) *Topology of LSTM*: LSTM is a network designed to encode contextual information of a temporal sequence with feedback loops. It contains cycles that feed the network activations from a previous time-step to influence predictions at the current time-step [33]. The unfolded chain structure of LSTM in an input sequence  $[\mathbf{f}_1, \mathbf{f}_2 \dots \mathbf{f}_k]$  is illustrated in Fig. 3 [34], where  $\mathbf{h}_j$  ( $j = 1, 2 \dots k$ ) is the hidden state at time-step  $j$  and  $\mathbf{c}_j$  is the activation vector. In the recurrent regression, the LSTM unit uses previous state  $(\mathbf{h}_{j-1}, \mathbf{c}_{j-1})$  and current feature  $\mathbf{f}_j$  to update current state  $(\mathbf{h}_j, \mathbf{c}_j)$  and compute wrist angle  $y_j$ . In this way the historical information can be passed recursively in the whole loop of LSTM.

2) *Update of LSTM Units*: Basic elements of LSTM include an input gate to control activations for the memory cell, a forget gate to drop useless information of the past cell status, and an output gate to control the output activations for the ultimate state. The update of LSTM units at time-step  $j$  can be described [35]

$$\begin{aligned}
 \mathbf{i}_j &= \delta(\mathbf{W}_i[\mathbf{h}_{j-1}, \mathbf{f}_j] + \mathbf{b}_i) \\
 \mathbf{m}_j &= \delta(\mathbf{W}_m[\mathbf{h}_{j-1}, \mathbf{f}_j] + \mathbf{b}_m) \\
 \mathbf{o}_j &= \delta(\mathbf{W}_o[\mathbf{h}_{j-1}, \mathbf{f}_j] + \mathbf{b}_o) \\
 \mathbf{c}_j &= \mathbf{i}_j \odot \tanh(\mathbf{W}_c[\mathbf{h}_{j-1}, \mathbf{f}_j] + \mathbf{b}_c) + \mathbf{m}_j \odot \mathbf{c}_{j-1} \\
 \mathbf{h}_j &= \mathbf{o}_j \odot \tanh(\mathbf{c}_j) \\
 y_j &= \mathbf{W}_y \mathbf{h}_j + \mathbf{b}_y
 \end{aligned} \tag{1}$$

where  $\mathbf{i}_j$  is the input gate,  $\mathbf{m}_j$  is the forget gate,  $\mathbf{o}_j$  is the output gate,  $\delta$  is the logistic sigmoid function,  $\mathbf{W}$  is the weight matrix in each gate and layer,  $\mathbf{b}$  is the corresponding bias vector and  $\odot$  is the scalar product. The initial state  $(\mathbf{h}_0, \mathbf{c}_0)$  will be settled after model training for subsequent predictions.

### C. Training of CNN-LSTM

In this study we adopt the idea of separate training following the approach in reference [36]. Specifically, the tuning of CNN and LSTM is conducted in two subsequent steps. Firstly, a regression layer is attached to the presented CNN architecture to complete a supervised learning. In this step, the model inputs are sEMG matrices and observations are wrist angles. Secondly, deep feature vectors are extracted from a fully connected layer of the well-trained CNN, based on which feature sequences are constructed to train LSTM for sequence regression. Different from structures such as Long-term Recurrent Convolutional Networks (LRCNs) which trains CNN

and LSTM jointly [37], our model can be more efficient in model training since 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 [36].

1) *Training Setting of CNN*: Hyper-parameters of presented CNN are mainly identified referring to pilot studies in PR schemes [15] and then determined via empirical manual tuning. As a general setting in this study, the network is trained in a 128-sized mini-batch as employed in [19] for 50 epochs by stochastic gradient descent with momentum (SDGM). The dynamic learning rate of CNN is 0.0001 in initialization and drops 90% after every 10 epochs. The slope scale is set as 0.1 in all leaky ReLU layers. The dropout rate in each dropout layer is 30%. Other training strategies follow default settings in Matlab 2018b.

2) *Training Setting of LSTM*: In our study the time duration of a regression sequence is set to be 1 second. This achieves a trade-off between the information quantity of temporal dependencies and computational loads in practical implementation. LSTM is trained in a 64 sized mini-batch for 100 epochs via adaptive moment estimation (ADAM). The dynamic learning rate is initialised to be 0.001 and drops 90% after every 10 epochs. Since LSTM is prone to over-fitting more easily than conventional recurrent neural networks, herein only one LSTM layer with 50 hidden units is adopted. A dropout layer with 30% dropout rate is added for regularization.

## III. MATERIALS AND EXPERIMENTAL METHODS

### A. Experiment Setup

Approved by the Mathematics, Physical Science and Engineering joint Faculty Research Ethics Committee of University of Leeds, UK (reference MEEC 18-006), six healthy subjects (five males and one female, aged 24-30) took part in the experiment. The written informed consent was obtained from each subject before data collection. Following Fig. 4 (a), 12 bipolar electrodes were placed on the proximal portion of the forearm to collect sEMG signals in 6 channels. Reference electrodes were placed near the wrist. The inter-electrode distance in the proximal-distal direction was around 20 mm for reducing the crosstalk effect.

As shown in Fig. 4 (b), in experiments participants were asked to perform four pre-defined wrist movement protocols. They were allowed to quit the experiments in case of any discomfort. The tested hand should be kept in a relaxing state to avoid muscle fatigue, with the upper limb supported vertically on the desk and the palm facing inside. All motions started from this rest position. Each protocol consisted of 3 sub-trials/sessions, and each session was composed of continuous wrist movements lasting around 3 minutes. A detailed description is reported in Table I.

From Table I we can see that in P1-P3 only one DoF of the wrist motions was activated to complete single-DoF tasks. On the contrary, P4 aimed at multi-DoF tasks and all 3 DoFs were involved simultaneously. Obviously, P4 is naturally more complex and challenging compared with P1-P3 [38], but it bears closer similarity with real-life movements [8] and can



Fig. 4: Experiment setup [31]. (a) Electrodes placement. (b) Data acquisition.

TABLE I: List of Performed Contractions

Protocol	Description	Active DoF
P1	Sinusoidal contractions	Flexion-extension (F-E)
P2	Sinusoidal contractions	Pronation-supination (P-S)
P3	Sinusoidal contractions	Radial-ulnar deviation (R-U)
P4	Co-contractions of the wrist	F-E+P-S+R-U

speed up the training process. The frequency of sinusoidal contractions was around 0.1 Hz, meaning that a cycle of wrist rotation (such as rest-flexion-rest-extension-rest in P1) was about 10 seconds.

In this study an attitude heading reference system (AHRS), composed of a tri-axial accelerometer, gyroscope and magnetometer, was utilized to obtain hand orientation [39]. Wrist angles, which worked as the ground-truth in supervised learning, were calculated based on Euler angles from AHRS. Referring to Fig. 4 (b), both sEMG signals and wrist movements were recorded simultaneously with Shimmer wearable sensors [40] attached on the back of the testing hand. Sampling rates for accelerometer, gyroscope, magnetometer and sEMG were set as 100 Hz, 100 Hz and 75 Hz and 1024 Hz respectively. The online data streaming was implemented in a home-made software based on Shimmer Matlab Instrument Driver [40].

### B. Data Pre-processing

In our experiments sEMG signals were processed using a 3<sup>rd</sup> order Butterworth high pass filter (20 Hz) to remove movement artifacts [41] and a low pass filter (450 Hz) to remove unusable high frequency noise [42]. A Min-Max scaling was applied to normalize sEMG in each channel [43]. As for data segmentation, the analysis window was set to be 100ms with increment of 50 ms. Thus the size of sEMG matrix ( $1 \times L \times N$ ) was  $1 \times 101 \times 6$  in our experiments. Since the time duration of a feature sequence was set to be 1 second empirically, there were 18 time-steps in  $[\mathbf{f}_1, \mathbf{f}_2 \dots \mathbf{f}_k]$ , i.e.  $k = 18$ .

### C. Model Evaluation

The analysis of sEMG-based wrist kinematics estimation was composed of intra-session and inter-session evaluations. To implement intra-session evaluations, the data in one session/trial of each protocol was split into four folds, where the first three were used for model training and the last for testing. To avoid data leakage, the splitting should be conducted before data pre-processing. In inter-session evaluations one whole session was used for model training and another session in the same protocol was used for testing. This method could better

validate the model robustness against time-dependent changes of sEMG signals. The number of training samples extracted by sliding windows is around 1500-2000 for intra-session evaluations and 2000-3000 for inter-session evaluations.

In this study we used the coefficient of determination ( $R^2$ ) [44] as the metric to quantify the regression performances. The mathematical expression of  $R^2$  is

$$R^2 = 1 - \frac{\text{Var}(\boldsymbol{\alpha}^d - \mathbf{y}^d)}{\text{Var}(\boldsymbol{\alpha}^d)} \quad (2)$$

where  $\boldsymbol{\alpha}^d$  are measured wrist angles by the sensor in  $d$ th DoF and  $\mathbf{y}^d$  are model estimations. According to Eq. (2), the numerator of  $R^2$  is the mean squared error (RMSE) which is normalized by the variance of correct labels in the denominator [5]. Compared with RMSE,  $R^2$  is more robust to the numerical range of labels.  $R^2$  at perfect estimation is equal to one, whilst a negative value means that estimation errors are larger than the variance of target values.

## IV. EXPERIMENTAL RESULTS

### A. Visual Exploration of sEMG Features

Visual exploration allows intuitive analysis of the distributions or potential correlations between certain variables. In this section, t-Distributed Stochastic Neighbour Embedding (t-SNE) is utilized to project extracted CNN features (in testing sets) into two principal dimensions for visualization [45]. For comparison, a widely applied temporal-spatial feature set [46–48] consisting of mean absolute value (MAV), root mean square (RMS), variance (VAR) and 4<sup>th</sup> order autoregressive coefficients (4<sup>th</sup> AR) are calculated. Scatter plots of projected sEMG features in P1 (F-E), P4 (F-E) and P4 (P-S) of intra-session evaluations are shown in Fig. 5, where the two axes represent two principal features, respectively. The angles of scatters (features) are reflected in parula colormap, with the pure yellow representing the positive maximal values in one DoF and pure purple for the negative maximum.

From Fig. 5 we can see that in each dataset the clustering of scatters projected from CNN features is significantly better than that of hand-crafted features. In the left part of each sub-figure, scatters with similar colour are gathering whilst those with different colours are highly distinguishable. On the contrary, scatters in the right one are overlapped heavily, even among the yellow ones and the blue ones. Compared with P1 (F-E), the clustering of scatters becomes worse for hand-crafted features in P4 (F-E). This deterioration becomes more evident in P4 (P-S), where distributions among scatters from CNN features become also ambiguous. A possible reason for the deterioration is that the crosstalk of sEMG can be quite serious in multi-DoFs tasks due to our forearm anatomy [46]. Since muscle fibres of extensors and flexors are much thicker and also located in a more superficial layer of the forearm, information of other DoFs are easier to be buried in compounded sEMG.

### B. Intra-session Estimations in Single-DoF Tasks

Fig. 6 shows wrist angles captured by AHRS system in P1-P3 of Subject 5 together with estimations of CNN and CNN-

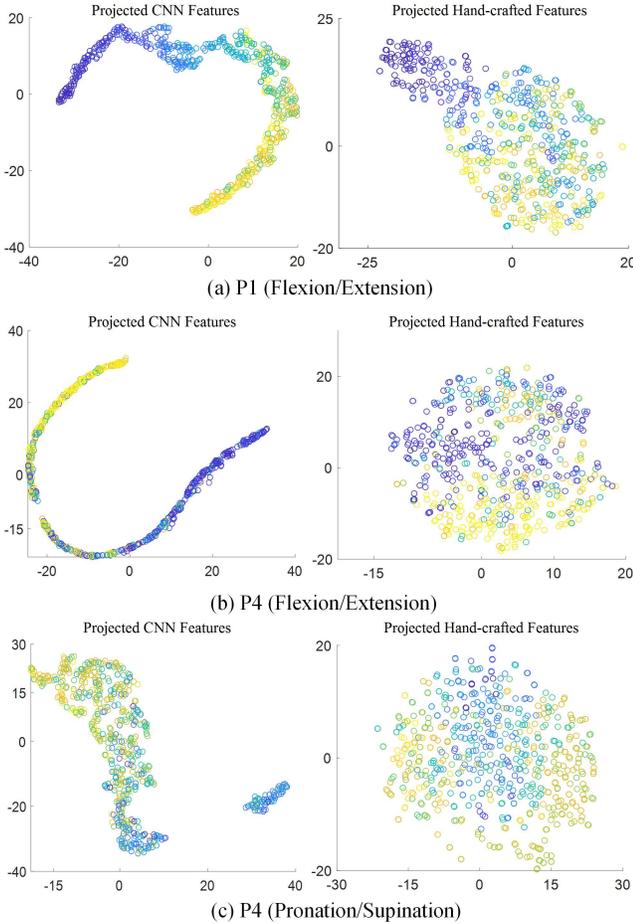


Fig. 5: Distribution of CNN features and hand-crafted features in testing sets of Subject 5 after dimension reduction. Scatters in (a)-(c) correspond to features from P1(F-E), P4(F-E) and P4(P-S), respectively.

LSTM. As illustrated in the figure, trajectories reconstructed by CNN-LSTM (in red) are smoother and much closer to the ground-truth (in blue) than CNN trajectories (in yellow) in all tasks. This is because the history information of successive deep feature vectors in a sequence is further exploited by CNN-LSTM, which improves estimation accuracies significantly. Another interesting result is that the estimated trajectories of both CNN and CNN-LSTM in P1 are better than their corresponding results in P2 and P3. As shown in visual exploration, feature scatters in the F-E DoF are much more distinguishable than those in the other two DoFs.

In this section two representative ML models, i.e. SVR and random forest (RF), are implemented to compare with DL techniques. SVR can project sEMG features into a higher dimensional space via kernel functions, whilst RF is currently the most popular ensemble learning technique. The outperformance of SVR and RF over other shallow models such as LR and ANN have been verified in pilot studies [5, 7]. To evaluate the effectiveness of CNN features for sequential learning, the conventional LSTM with hand-crafted features is also applied for comparison. Same to visual exploration, MAV, RMS, VAR and 4<sup>th</sup> AR are working as hand-crafted features for ML models. To be consistent with CNN-LSTM, we have reduced the dimension of hand-crafted features to 20 using Principle

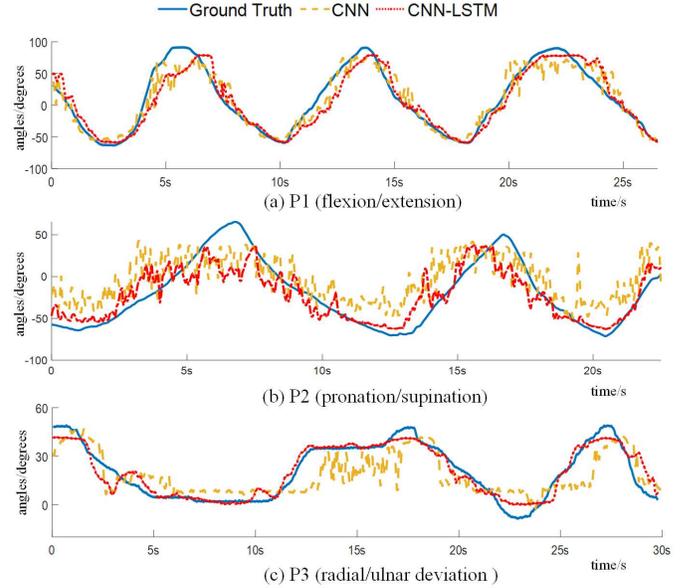


Fig. 6: Wrist motions and intra-session estimations of CNN and CNN-LSTM for P1-P3 in Subject 5. The mean absolute error (MAE) of two models are (a) 13.14 for CNN and 12.42 for CNN-LSTM; (b) 23.88 for CNN and 20.03 for CNN-LSTM; (c) 9.16 for CNN and 3.39 for CNN-LSTM. It is noted that MAE of different DoFs are not comparable due to different scales.

TABLE II:  $R^2$  of SVR, RF, CNN, LSTM and the proposed hybrid model in Single-DoF Tasks (P1-P3) of Intra-session Evaluations.

Subjects	Protocols	SVR	RF	CNN	LSTM	Proposed
1	P1(F-E)	0.56	0.71	0.56	0.84	<b>0.92</b>
	P2(P-S)	0.26	0.28	0.32	0.53	<b>0.65</b>
	P3(R-U)	0.56	0.59	0.66	0.67	<b>0.87</b>
2	P1(F-E)	0.60	0.69	0.66	0.81	<b>0.85</b>
	P2(P-S)	0.37	0.48	0.45	0.53	<b>0.56</b>
	P3(R-U)	0.22	0.25	0.31	0.46	<b>0.64</b>
3	P1(F-E)	0.35	0.38	0.42	0.7	<b>0.80</b>
	P2(P-S)	0.46	0.63	0.58	0.75	<b>0.83</b>
	P3(R-U)	0.18	0.19	0.22	0.52	<b>0.56</b>
4	P1(F-E)	0.35	0.40	0.41	0.67	<b>0.75</b>
	P2(P-S)	0.17	0.16	0.21	0.34	<b>0.46</b>
	P3(R-U)	0.40	0.48	0.43	0.66	<b>0.88</b>
5	P1(F-E)	0.84	0.86	0.84	0.82	<b>0.91</b>
	P2(P-S)	0.51	0.52	0.62	0.65	<b>0.71</b>
	P3(R-U)	0.59	0.71	0.67	0.80	<b>0.90</b>
6	P1(F-E)	0.71	0.76	0.74	0.81	<b>0.91</b>
	P2(P-S)	0.21	0.30	0.36	0.42	<b>0.64</b>
	P3(R-U)	0.36	0.32	0.40	0.51	<b>0.69</b>
Average	P1(F-E)	0.57	0.63	0.61	0.78	<b>0.86</b>
	P2(P-S)	0.33	0.40	0.42	0.54	<b>0.64</b>
	P3(R-U)	0.39	0.42	0.45	0.60	<b>0.76</b>

Component Analysis (PCA). Following previous studies [5], a radial basis function (RBF) is adopted for SVR. Besides, the hyper-parameters of SVR and RF are optimized via the 5-fold inner cross-validation.

Table II summarizes intra-session performances of SVR, RF, CNN, LSTM and CNN-LSTM in P1-P3 of Subject 1-6. As we can see, the presented hybrid model outperforms other models in all trials of all protocols. The outperformance can be more evident in some datasets, such as P2 and P3 in nearly all participants. In addition, by exploiting the correlations among adjacent sEMG samples, performances of conventional LSTM

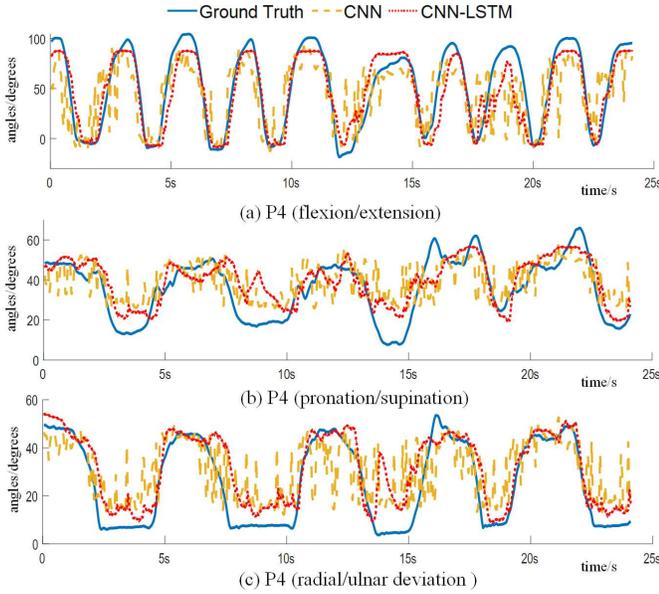


Fig. 7: Wrist motions and intra-session estimations of CNN and CNN-LSTM in P4 of Subject 5. MAE of two models are (a) 21.08 for CNN and 12.85 for CNN-LSTM; (b) 9.52 for CNN and 7.61 for CNN-LSTM; (c) 9.89 for CNN and 6.75 for CNN-LSTM.

are also better than SVR/RF/CNN in most cases. In protocol P1, the average  $R^2$  values of LSTM and CNN-LSTM are closer, whereas in other protocols LSTM is evidently inferior to CNN-LSTM. A possible reason is the deterioration of hand-crafted features can be more serious than CNN features in these two DoFs (P-S and R-U). A more detailed verification can be found in the visual exploration (Section IV.A). Besides, the conventional single-stream CNN is in general comparable to SVR and RF in sEMG-based wrist kinematics estimation. This result is similar to pilot studies in pilot studies [15].

### C. Intra-session Estimations in Multi-DoF Tasks

Different from single-DoF tasks (P1-P3), the multi-DoF task (P4) requires co-activations of 3 DoFs. Fig. 7 demonstrates the intra-session estimations of CNN and CNN-LSTM in P4 of Subject 5. In accordance with single-DoF tasks, the reconstructed trajectories of CNN-LSTM are much closer to the ground-truth in all DoFs. As for  $R^2$  values, CNN-LSTM reaches higher scores than other four models, indicating an evident improvement in model accuracy.  $R^2$  values of each DoF in six subjects are listed in Table III. Same to P1-P3, performances of CNN, RF and SVR are in general close to each other, whilst LSTM outperforms these non-recurrent regression techniques in most cases. Consistent with results in visual exploration, deteriorations in estimation accuracies can be found in each DoF of P4 compared with those in P1-P3, indicating that the features of samples become harder to recognize in the multi-DoF tasks.

### D. Inter-session Estimations in Single/Multiple DoFs Tasks

Fig. 8 illustrates the inter-session performance of CNN and CNN-LSTM in P1-P3 of Subject 5. Performances of both CNN and CNN-LSTM become a little bit worse compared to

TABLE III:  $R^2$  of SVR, RF, CNN, LSTM and the proposed hybrid model in Multi-DoF Tasks (P4) of Intra-session Evaluations.

Subjects	Protocols	SVR	RF	CNN	LSTM	Proposed
1	F-E	0.44	0.52	0.55	0.78	<b>0.87</b>
	P-S	0.30	0.31	0.30	0.51	<b>0.58</b>
	R-U	0.40	0.38	0.39	0.61	<b>0.69</b>
2	F-E	0.63	0.63	0.62	0.73	<b>0.82</b>
	P-S	0.19	0.27	0.28	0.39	<b>0.47</b>
	R-U	0.37	0.36	0.40	0.48	<b>0.61</b>
3	F-E	0.35	0.44	0.46	0.67	<b>0.70</b>
	P-S	0.37	0.39	0.50	0.49	<b>0.70</b>
	R-U	0.15	0.30	0.27	0.36	<b>0.42</b>
4	F-E	0.51	0.43	0.44	0.62	<b>0.67</b>
	P-S	0.21	0.23	0.25	0.40	<b>0.40</b>
	R-U	0.53	0.52	0.55	0.65	<b>0.73</b>
5	F-E	0.49	0.54	0.50	0.81	<b>0.86</b>
	P-S	0.31	0.37	0.40	0.59	<b>0.65</b>
	R-U	0.44	0.49	0.54	0.70	<b>0.83</b>
6	F-E	0.63	0.69	0.73	0.78	<b>0.89</b>
	P-S	0.25	0.29	0.34	0.43	<b>0.53</b>
	R-U	0.66	0.65	0.55	0.68	<b>0.74</b>
Average	F-E	0.51	0.54	0.55	0.73	<b>0.80</b>
	P-S	0.27	0.31	0.35	0.47	<b>0.56</b>
	R-U	0.43	0.45	0.45	0.58	<b>0.67</b>

intra-session evaluations in Fig. 6 due to domain shifts among different sessions, but the curves reconstructed by CNN-LSTM still manage to match the ground-truth. Fig. 9 illustrates comparisons among all regression techniques following P1-P4, in which the outperformance of CNN-LSTM are still evident. As for wrist motions in flexion and extension,  $R^2$  values of CNN-LSTM can be as high as 0.93 and 0.74 in new testing sessions of P1 and P4 (F-E), respectively, indicating a reliable proportional myoelectric control in this DoF. Besides, promising accuracies can be achieved by ML models in P1 (SVR and RF reach 0.73 and 0.79, respectively). As is discussed in visual exploration, the higher accuracies in F-E are mainly caused by the upper limb anatomy, which on the other hand leads to non-negligible cross-talks for sEMG of other two DoFs.

### E. Comparison of Time-steps in CNN-LSTM

The time-step  $k$  in feature sequence  $[\mathbf{f}_1, \mathbf{f}_2 \dots \mathbf{f}_k]$  determines the number of sEMG samples to be included in sequence regression. A larger  $k$  denotes a longer term of time dependencies which may contribute to a higher accuracy but also results in a heavier computational load. In this subsection four different time-steps are evaluated for CNN-LSTM, i.e. 8, 18, 58, 98 for the value of  $k$  which correspond to 0.5s, 1s, 3s and 5s in time duration, respectively. Estimation results in inter-session evaluations are illustrated in Fig. 10. In general, the  $R^2$  of CNN-LSTM improves gradually along with the increase of time-steps, indicating that the exploitation of long-term time dependencies contributes to a higher estimation accuracy in most scenarios. Empirically, a sequence in 1s duration can reach a compromise in model effectiveness and efficiency. Besides, a too large sequence is inapplicable for real-time myoelectric control since the intention prediction is expected to be implemented without evident time delays.

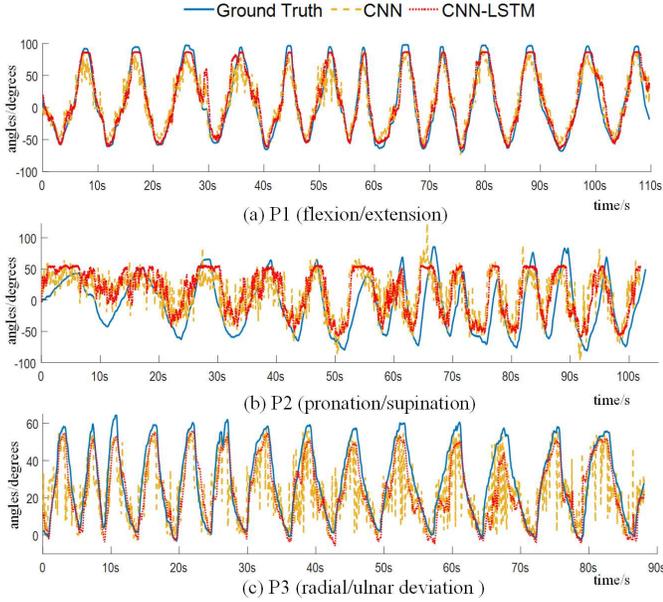


Fig. 8: Inter-session estimations of CNN and CNN-LSTM following P1-P3 of Subject 5. MAE of two models are (a) 14.93 for CNN and 10.22 for CNN-LSTM; (b) 31.70 for CNN and 28.0 for CNN-LSTM; (c) 9.56 for CNN and 8.11 for CNN-LSTM.

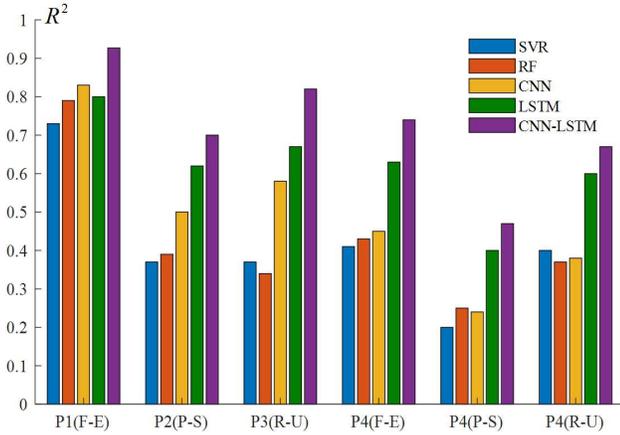


Fig. 9: Inter-session evaluations of SVR, RF, CNN, LSTM and the proposed CNN-LSTM in P1-P4.

### F. Comparison of sEMG Matrices

Besides the architecture and hyper-parameters, sEMG input matrices also have a non-negligible impact on CNN-based feature extraction and can then influence the estimation accuracy of CNN-LSTM. In Section II, we obtain the spectrum-based sEMG matrices by applying FFT on each sliding window. A more intuitive method is to construct matrices in the time domain directly. The comparison of CNN and CNN-LSTM with temporal and spectral sEMG matrices in intra-session evaluations can be found in Fig. 11. For simplicity, CNN/CNN-LSTM with temporal or spectral inputs are shorted as CNN<sub>t</sub>, CNN<sub>s</sub>, CNN-LSTM<sub>t</sub> and CNN-LSTM<sub>s</sub>, respectively. It can be observed that CNN<sub>s</sub> outperforms CNN<sub>t</sub> in all protocols, which contributes to the outperformance of CNN-LSTM<sub>s</sub> over CNN-LSTM<sub>t</sub> accordingly. This superiority becomes more significant in multi-DoF tasks. A possible reason is that the

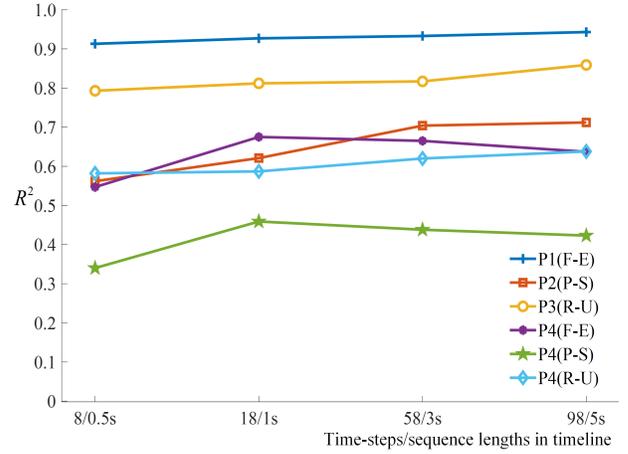


Fig. 10: Comparison between time-steps/sequence lengths of CNN-LSTM in inter-session evaluations.

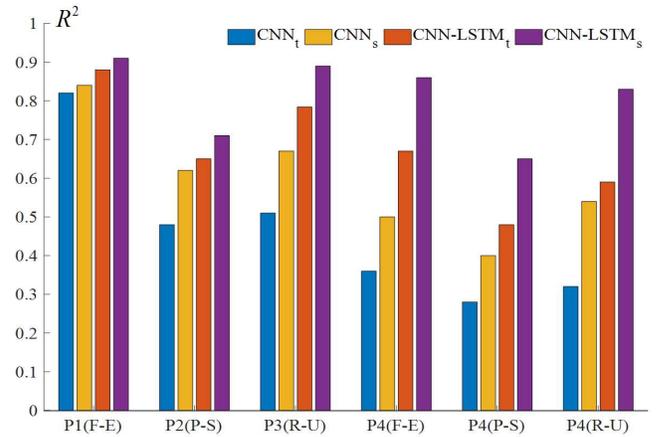


Fig. 11: Intra-session evaluations of CNN and CNN-LSTM with two types of sEMG matrices. CNN/CNN-LSTM with temporal or spectral inputs are shorted as CNN<sub>t</sub>, CNN<sub>s</sub>, CNN-LSTM<sub>t</sub> and CNN-LSTM<sub>s</sub>, respectively.

sEMG collected by sparse electrodes can be regarded as the superimposition of signals from multiple muscles. During voluntary contractions, the firing rates of motoneuron in these muscles are different [49], thus the spectrum information can be more representative and distinguishable.

### G. Comparison of Deep Feature Dimensions

As shown in Fig. 2, the sEMG matrix is imported into a pre-trained CNN and the vector extracted from the 2<sup>nd</sup> FC Block works as the CNN feature for sequential learning in each time-step. Since there are 20 hidden units in this layer, the dimension of deep features is 20. Compared with CNN architectures in many previous studies [15, 16, 21], the dimension of our last FC Block is smaller since we empirically found that a too large dimension might not be able to benefit the performances of CNN and CNN-LSTM significantly. TABLE IV illustrates the  $R^2$  values of CNN and CNN-LSTM when using different number of hidden units in the 2<sup>nd</sup> FC layer, which indicates that in our experiments 20 dimensions can be a good choice for both two models.

TABLE IV:  $R^2$  of CNN and CNN-LSTM when using different number of hidden units in the 2<sup>nd</sup> FC layer of CNN.

Protocols	Models	Number of hidden units in the 2 <sup>nd</sup> FC layer					
		2	5	10	20	50	100
P1 (F-E)	CNN	0.40	0.66	0.78	0.82	0.83	0.81
	CNN-LSTM	0.79	0.85	0.88	0.93	0.91	0.91
P2 (P-S)	CNN	0.33	0.41	0.45	0.48	0.49	0.51
	CNN-LSTM	0.60	0.65	0.68	0.69	0.70	0.69
P3 (R-U)	CNN	0.42	0.48	0.51	0.57	0.60	0.61
	CNN-LSTM	0.77	0.80	0.80	0.82	0.83	0.84
P4 (F-E)	CNN	0.32	0.38	0.40	0.44	0.43	0.44
	CNN-LSTM	0.70	0.73	0.74	0.76	0.75	0.77
P4 (P-S)	CNN	0.15	0.22	0.24	0.25	0.23	0.26
	CNN-LSTM	0.40	0.42	0.46	0.45	0.44	0.48
P4 (R-U)	CNN	0.25	0.31	0.34	0.37	0.37	0.39
	CNN-LSTM	0.63	0.65	0.67	0.68	0.67	0.68

## V. DISCUSSION

To further improve regression accuracies and robustness, both CNN and LSTM are now becoming prevalent in sEMG-based motion estimation. In this paper, we presented a hybrid model to combine these two techniques, i.e. CNN for automatic feature extraction and LSTM for sequential regression, such that the temporal-spatial correlations in sEMG signals can be extracted more efficiently. Conventional ML techniques rely deeply on manual feature extraction and selection. This process requires good domain knowledge or experience, and useful information may be easily buried in hand-crafted features. On the contrary, CNN extracts features from raw sEMG directly and automatically by learning the signal characteristics via layer-by-layer processing. The convolution operations also enables CNN to extract spatial correlations of sEMG signals from multi-channels. As mentioned in many previous works, CNN features can be useful to represent patterns of muscle activations, and the automatic feature extraction can help to reduce the information loss. Visual explorations of two types of features (details can be found in Section IV.A) indicate that distributions of CNN features can be better correlated with wrist motions than many hand-crafted features. Therefore, the CNN-LSTM can outperform conventional LSTM which only uses hand-crafted features.

Secondly, SVR, RF and CNN are all non-recurrent models, which inherently ignore the temporal dependencies of successive sEMG samples. In fact, during continuous muscle contractions there are supposed to be strong temporal-dependencies in sEMG signals. Thus it is reasonable to consider sEMG as time-series data in regression tasks. In this study the feature sequences  $[\mathbf{f}_1, \mathbf{f}_2 \dots \mathbf{f}_k]$  are reconstructed for LSTM to further exploit the history information of successive deep feature vectors. From previous literatures [23] it can be inferred that the recurrent networks such as LSTM have shown superiority to many non-recurrent models. Our experiment results also demonstrate the outperformance of recurrent architectures. Therefore, by efficiently extracting the temporal-spatial correlations in sEMG signals, CNN-LSTM further improves regression accuracies in both single and multiple DoF tasks.

A main limitation of our current method is the model generalization in multi-days and multi-subjects. Due to the non-stationary characteristics of sEMG signals, it is reported

that classification/regression performances could decrease substantially over time [50]. Besides, sEMG signals have a user-dependent nature, causing recordings to differ even when signals are measured from the same location with the same motion. Therefore, a pre-trained model may not be able to perform accurately in a new subject. These issues can be summarized as the domain shift problems in machine learning applications, since data-driven methods rely on the assumption that training and testing data should stem from same underlying distributions. To this end we will further investigate domain/rule adaptation approaches to improve the generalization of CNN-LSTM. Besides, more volunteers are going to be recruited for a better verification of our method.

## VI. CONCLUSION

In this paper, we presented a hybrid model to combine CNN-based feature extraction and LSTM-based sequence regression in wrist kinematics estimation, which could extract temporal-spatial correlations in sEMG efficiently. Through visual exploration, we verified that deep features extracted by CNN were more representative than traditional hand-crafted features. By exploiting contextual information in deep features, the presented CNN-LSTM outperformed conventional CNN, LSTM as well as representative ML approaches in both intra-session and inter-session evaluations.

## REFERENCES

- [1] J. M. Hahne, M. A. Schweisfurth, M. Koppe, and D. Farina, "Simultaneous control of multiple functions of bionic hand prostheses: Performance and robustness in end users," *Science Robotics*, vol. 3, no. 19, p. eaat3630, 2018.
- [2] S. Pasinetti, M. Lancini, I. Bordini, and F. Docchio, "A novel algorithm for emg signal processing and muscle timing measurement," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 11, pp. 2995–3004, 2015.
- [3] 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 transactions on biomedical circuits and systems*, vol. 13, no. 3, pp. 516–528, 2019.
- [4] A. W. Shehata, E. J. Scheme, and J. W. Sensinger, "Evaluating internal model strength and performance of myoelectric prosthesis control strategies," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, no. 5, pp. 1046–1055, 2018.
- [5] J. M. Hahne, F. Biessmann, N. Jiang, H. Rehbaum, D. Farina, F. Meinel, K.-R. Müller, and L. Parra, "Linear and nonlinear regression techniques for simultaneous and proportional myoelectric control," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, no. 2, pp. 269–279, 2014.
- [6] A. Ameri, E. N. Kamavuako, E. J. Scheme, K. B. Englehart, and P. A. Parker, "Support vector regression for improved real-time, simultaneous myoelectric control," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, no. 6, pp. 1198–1209, 2014.
- [7] A. Ameri, E. J. Scheme, K. B. Englehart, and P. A. Parker, "Bagged regression trees for simultaneous myoelectric force estimation," in *2014 22nd Iranian Conference on Electrical Engineering (ICEE)*. IEEE, 2014, pp. 2000–2003.
- [8] K. Bakshi, M. Manjunatha, and C. Kumar, "Estimation of continuous and constraint-free 3 dof wrist movements from surface electromyogram signal using kernel recursive least square tracker," *Biomedical Signal Processing and Control*, vol. 46, pp. 104–115, 2018.
- [9] J. Liu, Y. Ren, D. Xu, S. H. Kang, and L.-Q. Zhang, "Emg-based real-time linear-nonlinear cascade regression decoding of shoulder, elbow and wrist movements in able-bodied persons and stroke survivors," *IEEE Transactions on Biomedical Engineering*, 2019.
- [10] R. N. Khushaba, A. H. Al-Timemy, A. Al-Ani, and A. Al-Jumaily, "A framework of temporal-spatial descriptors-based feature extraction for improved myoelectric pattern recognition," *IEEE Transactions on Neural*

- Systems and Rehabilitation Engineering*, vol. 25, no. 10, pp. 1821–1831, 2017.
- [11] S. Thongpanja, A. Phinyomark, F. Quaine, Y. Laurillau, C. Limsakul, and P. Phukpattaranont, “Probability density functions of stationary surface emg signals in noisy environments,” *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 7, pp. 1547–1557, 2016.
  - [12] Z. Zhou and J. Feng, “Deep forest: Towards an alternative to deep neural networks. arxiv 2017,” *arXiv preprint arXiv:1702.08835*.
  - [13] Y. Zhao, H. Li, S. Wan, A. Sekuboyina, X. Hu, G. Tetteh, M. Piraud, and B. Menze, “Knowledge-aided convolutional neural network for small organ segmentation,” *IEEE journal of biomedical and health informatics*, 2019.
  - [14] K.-H. Park and S.-W. Lee, “Movement intention decoding based on deep learning for multiuser myoelectric interfaces,” in *2016 4th International Winter Conference on Brain-Computer Interface (BCI)*. IEEE, 2016, pp. 1–2.
  - [15] M. Atzori, M. Cognolato, and H. Müller, “Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands,” *Frontiers in neurorobotics*, vol. 10, p. 9, 2016.
  - [16] Y. Du, W. Jin, W. Wei, Y. Hu, and W. Geng, “Surface emg-based inter-session gesture recognition enhanced by deep domain adaptation,” *Sensors*, vol. 17, no. 3, p. 458, 2017.
  - [17] W. Wei, Y. Wong, Y. Du, Y. Hu, M. Kankanhalli, and W. Geng, “A multi-stream convolutional neural network for semg-based gesture recognition in muscle-computer interface,” *Pattern Recognition Letters*, 2017.
  - [18] Z. Ding, C. Yang, Z. Tian, C. Yi, Y. Fu, and F. Jiang, “semg-based gesture recognition with convolution neural networks,” *Sustainability*, vol. 10, no. 6, p. 1865, 2018.
  - [19] A. Ameri, M. A. Akhaee, E. Scheme, and K. Englehart, “Regression convolutional neural network for improved simultaneous emg control,” *Journal of neural engineering*, 2019.
  - [20] W. Yang, D. Yang, J. Li, Y. Liu, and H. Liu, “Emg dataset augmentation approaches for improving the multi-dof wrist movement regression accuracy and robustness,” in *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2018, pp. 1268–1273.
  - [21] 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 Transactions on Human-Machine Systems*, vol. 49, no. 5, pp. 411–420, 2019.
  - [22] F. Quivira, T. Koike-Akino, Y. Wang, and D. Erdogmus, “Translating semg signals to continuous hand poses using recurrent neural networks,” in *2018 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*. IEEE, 2018, pp. 166–169.
  - [23] T.-A. Teban, R.-E. Precup, E.-C. Lunca, A. Albu, C.-A. Bojan-Dragos, and E. M. Petriu, “Recurrent neural network models for myoelectric-based control of a prosthetic hand,” in *2018 22nd International Conference on System Theory, Control and Computing (ICSTCC)*. IEEE, 2018, pp. 603–608.
  - [24] Y. He, O. Fukuda, N. Bu, H. Okumura, and N. Yamaguchi, “Surface emg pattern recognition using long short-term memory combined with multilayer perceptron,” in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2018, pp. 5636–5639.
  - [25] A. Samadani, “Gated recurrent neural networks for emg-based hand gesture classification. a comparative study,” in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2018, pp. 1–4.
  - [26] D. Biswas, L. Everson, M. Liu, M. Panwar, B. Verhoef, S. Patrika, C. H. Kim, A. Acharyya, C. Van Hoof, M. Konijnenburg *et al.*, “Cornet: Deep learning framework for ppg based heart rate estimation and biometric identification in ambulant environment,” *IEEE transactions on biomedical circuits and systems*, 2019.
  - [27] P. Xia, J. Hu, and Y. Peng, “Emg-based estimation of limb movement using deep learning with recurrent convolutional neural networks,” *Artificial organs*, vol. 42, no. 5, pp. E67–E77, 2018.
  - [28] D. Huang and B. Chen, “Surface emg decoding for hand gestures based on spectrogram and cnn-lstm,” in *2019 2nd China Symposium on Cognitive Computing and Hybrid Intelligence (CCHI)*. IEEE, 2019, pp. 123–126.
  - [29] 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, p. e0206049, 2018.
  - [30] S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry, “How does batch normalization help optimization?” in *Advances in Neural Information Processing Systems*, 2018, pp. 2483–2493.
  - [31] T. Bao, Z. Zhang, S. Zaidi, and S. Xie, “Surface-emg based wrist kinematics estimation using convolutional neural network,” in *Proceedings of BSN 2019*. IEEE, 2019.
  - [32] K. Chitturi and P. Onyisi, “How easily can neural networks learn relativity?” in *Journal of Physics: Conference Series*, vol. 1085, no. 4. IOP Publishing, 2018, p. 042020.
  - [33] U. Singh, J.-F. Determe, F. Horlin, and P. De Doncker, “Crowd forecasting based on wifi sensors and lstm neural networks,” *IEEE Transactions on Instrumentation and Measurement*, 2020.
  - [34] W. Bao, J. Yue, and Y. Rao, “A deep learning framework for financial time series using stacked autoencoders and long-short term memory,” *PLoS one*, vol. 12, no. 7, p. e0180944, 2017.
  - [35] B. Zhao, C. Cheng, Z. Peng, X. Dong, and G. Meng, “Detecting the early damages in structures with nonlinear output frequency response functions and the cnn-lstm model,” *IEEE Transactions on Instrumentation and Measurement*, 2020.
  - [36] 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 *Proceedings of the 23rd ACM international conference on Multimedia*. ACM, 2015, pp. 461–470.
  - [37] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, “Long-term recurrent convolutional networks for visual recognition and description,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 2625–2634.
  - [38] A. Ameri, E. N. Kamavuako, E. J. Scheme, K. B. Englehart, and P. A. Parker, “Real-time, simultaneous myoelectric control using visual target-based training paradigm,” *Biomedical Signal Processing and Control*, vol. 13, pp. 8–14, 2014.
  - [39] S. O. Madgwick, A. J. Harrison, and R. Vaidyanathan, “Estimation of imu and marg orientation using a gradient descent algorithm,” in *2011 IEEE international conference on rehabilitation robotics*. IEEE, 2011, pp. 1–7.
  - [40] A. Burns, B. R. Greene, M. J. McGrath, T. J. O’Shea, B. Kuris, S. M. Ayer, F. Strojescu, and V. Cionca, “Shimmer™—a wireless sensor platform for noninvasive biomedical research,” *IEEE Sensors Journal*, vol. 10, no. 9, pp. 1527–1534, 2010.
  - [41] C. J. De Luca, L. D. Gilmore, M. Kuznetsov, and S. H. Roy, “Filtering the surface emg signal: Movement artifact and baseline noise contamination,” *Journal of biomechanics*, vol. 43, no. 8, pp. 1573–1579, 2010.
  - [42] S. Micera, J. Carpaneto, and S. Raspopovic, “Control of hand prostheses using peripheral information,” *IEEE Reviews in Biomedical Engineering*, vol. 3, pp. 48–68, 2010.
  - [43] Z. Huang, C. Yang, X. Zhou, and T. Huang, “A hybrid feature selection method based on binary state transition algorithm and relief,” *IEEE journal of biomedical and health informatics*, 2018.
  - [44] A. d’Avella, A. Portone, L. Fernandez, and F. Lacquaniti, “Control of fast-reaching movements by muscle synergy combinations,” *Journal of Neuroscience*, vol. 26, no. 30, pp. 7791–7810, 2006.
  - [45] L. v. d. Maaten and G. Hinton, “Visualizing data using t-sne,” *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.
  - [46] A. Phinyomark, F. Quaine, S. Charbonnier, C. Serviere, F. Tarpin-Bernard, and Y. Laurillau, “Emg feature evaluation for improving myoelectric pattern recognition robustness,” *Expert Systems with applications*, vol. 40, no. 12, pp. 4832–4840, 2013.
  - [47] S. Muceli and D. Farina, “Simultaneous and proportional estimation of hand kinematics from emg during mirrored movements at multiple degrees-of-freedom,” *IEEE transactions on neural systems and rehabilitation engineering*, vol. 20, no. 3, pp. 371–378, 2011.
  - [48] B. Yu, X. Zhang, L. Wu, X. Chen, and X. Chen, “A novel postprocessing method for robust myoelectric pattern-recognition control through movement pattern transition detection,” *IEEE Transactions on Human-Machine Systems*, vol. 50, no. 1, pp. 32–41, 2019.
  - [49] K. Seki and M. Narusawa, “Firing rate modulation of human motor units in different muscles during isometric contraction with various forces,” *Brain research*, vol. 719, no. 1-2, pp. 1–7, 1996.
  - [50] 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 journal of biomedical and health informatics*, vol. 23, no. 4, pp. 1526–1534, 2018.