Analysis of the effect of the regulating parameters in recurrent deep neural networks on the regression of finger joint angles

Masoud Saheb Jameyan, Mohammad Ali Ahmadi Pajouh*, and Mohammad Hassan Moradi

Department of Biomedical Engineering, Amirkabir University of Technology, Tehran, Iran

*Correspondence e-mail: pajouh@aut.ac.ir.

Abstract

Accurate estimation of joint angles during limb movements plays a crucial role in the rehabilitation and diagnosis of neuromuscular and rheumatic disorders. This study aims to predict finger joint kinematics from surface electromyography (sEMG) signals using Long Short-Term Memory (LSTM) networks, which are well-suited for modeling temporal dependencies in physiological data. To enhance the model's generalization and reduce overfitting, four regularization strategies—LASSO, ridge, elastic net, and dropout—were systematically evaluated. Among these, LASSO and ridge regularization showed optimal performance when their coefficients were set to 0.0005, effectively balancing model complexity and prediction accuracy. While dropout was also beneficial, its performance declined at higher rates, with 0.2 identified as the most effective setting. The inclusion of appropriate regularization techniques led to a significant improvement in model accuracy, up to 20%, demonstrating their critical role in refining EMG-based kinematic estimation. The proposed LSTM model achieved a maximum prediction accuracy of 98% and an average of 96%, evaluated using the Pearson correlation coefficient. The results highlight the importance of selecting the appropriate regularization parameters to optimize both prediction accuracy and training speed in deep learning tasks designed to estimate joint angles.

Keywords: Regression**,** Deep Learning, Regularization, Finger joint angle, LSTM, sEMG.

1. Introduction

Stroke survivors often experience challenges with precise motor control, particularly in daily activities such as grasping objects [1]. Research has shown that robot-assisted rehabilitation can improve both short- and long-term motor control in the affected upper limb. However, there is limited evidence regarding how well these improvements transfer to real-world activities. Therefore, incorporating functional tasks such as object grasping into rehabilitation programs is essential to enhance practical recovery [2].

Additionally, surface electromyo graphy (sEMG) captures the bioelectrical signals produced by muscles during contraction, reflecting the overall electrical activity governed by the nervous system. Electrodes positioned on the skin's surface capture these signals, indicating the intention to move. Analyzing sEMG signals facilitates the identification of movement intentions and the assessment of muscle function.

In the domain of hand motion analysis, two primary challenges exist: classifying hand gestures and regressing of joint and limb movements. Shen et al. also used a convolutional neural network (CNN) to identify hand movements, utilizing two classifiers in succession [3]. Kahn et al. used a lightweight CNN for recognizing hand movements [4]. Asif et al. used a CNN to identify eleven wrist and finger movements, examining how changes in certain network parameters affected efficiency [5]. Chen et al. applied a 3D CNN to recognize finger movements, highlighting its advantage in capturing both the timing and shape of signals [6]. Nasri et al. employed a recurrent neural network with gated recurrent units to classify six distinct movements [7]. Simao et al. utilized deep recurrent networks to identify eight hand movements, achieving real-time classification [8]. Although gesture classification has been widely studied, accurately tracking continuous hand movements is still a major challenge. This study specifically addresses the regression of finger joint movements.

Previous studies aimed at mapping motor unit activity to finger kinematics have employed four methodological approaches: musculoskeletal modeling, traditional machine learning, conventional neural networks, and deep learning architectures. In the musculoskeletal modeling domain, He et al. estimated the metacarpophalangeal (MCP) joint angles of fingers using a muscle synergy model. They collected data using 32 surface electrodes [9]. Kim et al. established a muscle synergy and musculoskeletal model for estimating wrist joint angle [10]. Similarly, Zhang *et al.* employed fourteen electrodes alongside a Kalman filter algorithm to predict six finger joint angles [11]. Gilstrep *et al.* also adopted a musculoskeletal modeling strategy to continuously estimate the motion of the index finger [12]. In their study, Roy et al. utilized a 128-channel electrode array to decompose sEMG signals and predict joint angles [13]; however, this technique requires the experimental calibration of numerous biomechanical parameters, which poses significant limitations in terms of practical implementation and accuracy [14].

The second approach uses traditional machine learning. For example, Zhang *et al.* utilized a sparse pseudo-input gaussian process (SPGP) regression method to map EMG features to MCP joint angles [15]. Gao et al. utilized a least squares support vector machine that incorporates time-delayed features of sEMG signals to estimate continuous wrist palmar angles [16]. Nevertheless, these approaches often involve complex and labor-intensive feature extraction processes, which may result in the loss of critical signal information [17], [18].

The third category encompasses the use of conventional neural networks. Batayneh et al. used neural networks to create a nonlinear relationship between sEMG signals and hand joint angles [19]. Gao et al. developed a control method that utilizes a nonlinear autoregressive model with exogenous inputs (NARX) to estimate joint angles based on surface EMG signals [20].

Tacca et al. implemented a traditional neural network, combined with 150 electrodes, to estimate joint angles for four specific hand movements [21]. Despite showing promising results, this approach is highly dependent on the integration of numerous parameters and the availability of high-quality data [14].

Deep learning has resulted in significant advancements across various scientific fields [22, 23]. Recently, there has been an increasing exploration of deep neural networks for predicting finger kinematics, owing to their ability to learn features automatically and enhance generalization. Chen et al. utilized Long Short-Term Memory (LSTM) networks to forecast upper limb joint angles [24]. Kim et al. first reduced the dimensions of electromyogram signals and then applied a CNN to determine finger positions [25]. Qin et al. used a CNN to estimate the angle of the wrist joint [26]. Guo et al. also leveraged a convolutional framework for finger kinematics estimation [27]. RNNs have also been applied to estimate the position of limbs. Tang et al. employed LSTM to estimate the elbow and wrist joints [28]. Wen et al. introduced an CNN architecture to estimate neural drives during isometric contraction tasks [29]. Sun et al. proposed a multi-stream CNN for the continuous recognition of finger gestures [30]. Guo et al. forecasted finger kinematics by a multi-attention feature fusion network [31]. Wang et al. employed a hybrid unsupervised domain adaptation network to estimate joint angles for application in myoelectric control systems [32]. An et al. used convolutional layers transformer to estimate finger joint angles during grasping movements [33]. Wang et al. proposed an ensemble learning technique to estimate MCP joint angles during finger movements [34]. To estimate ten joint angles during six grasping tasks, Lin et al. adopted a network architecture combining inception and transformer models [35]. To estimate hand joint kinematics, Anam et al. initially performed dimensionality reduction on sEMG signals, followed by the implementation of an LSTM network [36].

Developing reliable and intuitive control systems for myoelectric prosthetic hands remains a key challenge in neurorehabilitation. While deep learning models have advanced in decoding hand kinematics from surface EMG signals, there are still notable methodological and practical limitations that need to be addressed. This study aims to bridge these gaps by emphasizing model optimization, signal efficiency, and physiological interpretability.

Previous studies have employed deep neural networks to estimate joint angles continuously; however, they often overlooked the impact of hyperparameter tuning on model performance. Moreover, the lack of structured regularization strategies may reduce the models' ability to generalize, particularly when working with datasets that contain limited repetitions.

Some existing approaches rely on using a large number of electrodes to achieve accurate results [13]. Although effective, this setup can limit real-world applicability due to its high cost, complexity, and discomfort for patients. Many current studies focus on estimating specific finger joints [9, 11, 12, 15, and 34] without considering the underlying neuromuscular factors that affect decoding performance. This oversight restricts physiological interpretability and limits the potential to provide valuable design feedback for optimizing electrode placement. Several earlier studies have integrated pre-processing and post-processing techniques to analyze sEMG signals and estimate joint angles [25, 36]. While these approaches can improve signal quality and reduce

noise, they often lead to increased computational complexity and longer processing times. The key contributions are as follows:

1. A new LSTM-based system with carefully adjusted regularization settings is proposed for estimating joint angles, achieving top accuracy (up to 98%, average: 96%) on the Ninapro dataset, demonstrating a novel approach to optimizing parameters in deep recurrent models.
2. A minimal-electrode configuration utilizing only 10 sEMG channels is proposed for precise estimation of multi-joint finger angles, marking a significant advancement for real-time clinical and wearable applications.
3. A joint-specific, muscle-level analysis framework has been developed to interpret the variations in accuracy across finger joints. This framework offers a novel perspective on the relationship between EMG, muscles, and joints, while also guiding future strategies for electrode placement.
4. A lightweight, end-to-end estimation pipeline has been designed that does not require complex pre- or post-processing. This enables real-time deployment with minimal computational overhead, representing a significant advancement in the practical application of embedded systems. The rest of the article introduces the materials and methods used, covering the dataset, LSTM, network architecture, and evaluation criteria. The third and fourth sections present the results and the subsequent discussion, respectively.

2. Methodes

2.1 Data set

This study employed data from the Ninapro Database 4 (DB4), which contains highly synchronized kinematic and sEMG recordings from the upper limbs of ten able-bodied subjects [37]. The cohort included six males and four females, with a mean age of 29.1 ± 3.9 years. All subjects were right-handed and reported no history of neuromuscular disorders. Participants were instructed to perform a range of standardized finger, hand, and wrist movements. Muscle activity associated with five primary flexion tasks was recorded via ten surface electrodes. Simultaneously, joint kinematics was captured using a CyberGlove II data glove, enabling precise monitoring of joint angles. Each movement is repeated six times [38]. Figure 1 shows the electrode placement on the forearm. Eight electrodes were evenly spaced around the forearm near the radiohumeral joint, just below the elbow. The remaining two electrodes were positioned over the forearm's main flexor and extensor muscles.

Figure 1: The placement of electrodes on arm muscles [38]

The finger flexion movements chosen for this study are illustrated in Figure 2 [38].

Figure 2: Finger flexion movements [38]

The hand's kinematic configuration was recorded using a CyberGlove II data glove equipped with 22 sensors, as shown in Figure 3. This glove was constructed from a lightweight, stretchable fabric with 22 integrated strain gauges. The device produced 22 8-bit values that correspond to the joint angles, with an average resolution of less than one degree, depending on the hand size of the wearer, the glove's fit, and the angular range of the specific joint [39].

Figure 3: The placement of bending sensors on finger joints

The sEMG signal processing procedure included filtering, synchronization, and relabeling. A Hampel filter was applied to remove the 50 Hz power-line interference and its harmonics from the sEMG signals. Inaccurate movement labels, caused by differences in human reaction times and experimental settings, were rectified using an expanded likelihood ratio technique. To achieve synchronization, each stream was up-sampled to the maximum sampling frequency using either nearest-neighbor or linear interpolation [39]. The study selected the root mean square (RMS) feature, as shown in equation (1).

$$RMS = \sqrt{1/N \sum_{i=1}^{N} emg_i^2} \qquad (1)$$

2.2. Structure of LSTM

Hochreiter and Schmidhuber introduced LSTM, an enhanced RNN architecture specifically designed to address the vanishing gradient problem found in traditional RNNs [40]. LSTM networks utilize gating mechanisms to overcome this issue. LSTM networks are highly effective for tasks such as prediction and classification because they can manage time series data with varying intervals between significant events. The structure of LSTM, illustrated in Figure 4, includes the input (x) and hidden state (h) at time step k and utilizes sigmoid (σ) and hyperbolic tangent (tanh) functions.

Figure 4: Structure of the LSTM

LSTM networks consist of three gates: the input gate, the output gate, and the forget gate. These gates allow the LSTM to selectively forget or retain new information in the memory cell, effectively capturing long-term dependencies. The input gate regulates how much new information is added to the memory cell, the forget gate decides which information should be discarded, and the output gate controls the flow of information from the memory cell to the hidden state. By employing these gates, LSTM networks can better handle long-term dependencies and enhance performance in sequence modeling tasks. LSTM's capability to tackle the gradient vanishing problem makes it well-suited for tasks involving sequential data, including natural language processing, speech recognition, and time series analysis. The operations of the LSTM layer can be described by the following equations [41].

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \quad (2)$$

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \quad (3)$$

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \quad (4)$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \quad (5)$$

$$y = o_t \circ \sigma_h(c_t) \quad (6)$$

These equations illustrate how the input gate, forget gate, cell state, output gate, and hidden state interact and depend on each other within the LSTM model. The parameters involved in these relationships are shown in Table 1.

Table 1: LSTM cell vectors and parameters [41]

2.3. Proposed LSTM Network Architecture

To model the temporal dynamics and long-term dependencies in EMG signals, an LSTM network was employed. LSTM cells are ideal for analyzing time-series data because they can remember information over long periods, filter out noise, and capture complex nonlinear patterns. This capacity makes them a strong option for estimating motion from EMG signals. The network receives time-series data as input, which is reshaped into 2D arrays—one dimension representing the time window length and the other representing the number of EMG channels (electrodes). The model processes this input using two stacked LSTM layers, each with 128 units and the hyperbolic tangent (tanh) activation function. Since each movement in the DB4 dataset repeats six times, two layers provide enough learning capacity. The network's output is a sequence of joint angles, recorded simultaneously with the EMG signals using bending sensors in a CyberGlove II. A fully connected (dense) layer follows the LSTM layers, which do not include any activation function.

The network was trained using the Adam optimizer [42] with a learning rate of 0.0005. The data were divided into training and testing sets at a 2:1 ratio. All experiments were conducted in Python using the TensorFlow library within the PyCharm development environment. To prepare the input matrices, a sliding window approach was used. Each window had a length of 150 milliseconds and moved along the time axis with a fixed step size (hop length) of 10 milliseconds. The block diagrams illustrating this design are shown in Figure 5.

Figure 5: Structure of proposed network

2.4. Regression Criterion

Three evaluation metrics were employed to evaluate the regression model's performance: root mean square error (RMSE), Pearson correlation coefficient (PCC), and box plots. The PCC is a statistical indicator that measures both the strength and direction of the linear relationship between actual values and predicted values. RMSE quantifies the average discrepancy between the model's predicted values and the actual values. Box plots were used to display the distribution of both predicted and real values. The criteria referenced are defined in equations 7 and 8, and the parameters governing these relationships are summarized in Table 2.

$$PCC = \frac{\sum_{i=1}^{n} MN}{\sqrt{\sum_{i=1}^{n} M^2}\sqrt{\sum_{i=1}^{n} N^2}} \quad (7)$$

$$M = y_{pred} - \bar{y}_{pred} \quad N = y_{true} - \bar{y}_{true}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{n} (y_{pred} - y_{true})^2} \quad (8)$$

Table 2: Parameters of regression criterion

2.5. Regularization

Regularization is a technique in machine learning and deep learning that reduces overfitting and enhances a model's generalization ability. It plays a crucial role in training neural networks by incorporating a penalty term into the loss function. This helps mitigate overfitting and promotes the learning of robust, meaningful features [43]. Regularization techniques, such as L1 and L2 regularization, dropout, and elastic net, improve model robustness and prediction accuracy. L2, L1, and elastic net limit parameter size, while dropout averages results over networks. These techniques improve performance and adaptability in models.

2.5.1. LASSO

LASSO regression, also known as L1 regularization, is a statistical modeling technique that balances simplicity and accuracy in estimating variable relationships and making predictions. LASSO regression, which stands for least absolute shrinkage and selection operator, adds a penalty term that encourages sparse solutions. This aids in feature selection and enhances model interpretability while also reducing overfitting and boosting the model's robustness and generalizability [44]. Expression 9 shows the cost function for LASSO regression, and Table 3 provides its parameters.

$$Cost\_function = Loss + \frac{\lambda}{2m} * \sum |w| \quad (9)$$

Table 3: Parameters of regularization methodes [31]

By constraining the sum of the absolute values, LASSO regularization encourages simpler models with fewer non-zero coefficients.

2.5.2. Ridge

L2 regularization, also referred to as weight decay, is synonymous with ridge regression. It drives the weights toward zero by incorporating a penalty term proportional to the sum of the squared values of the coefficients. This regularization technique stabilizes the model and prevents overfitting by reducing the magnitudes of the coefficients. L2 regularization encourages smaller weights, balancing model complexity and accuracy to produce more robust and generalized predictions [45]. The ridge regression cost function is defined as

$$Cost\_function = Loss + \frac{\lambda}{2m} * \sum |w|^2 \quad (10)$$

2.5.3. Elastic net

Elastic net offers a flexible regularization approach that allows for fine-tuning the balance between selecting important features and shrinking coefficient values. This makes it particularly powerful for high-dimensional datasets with many correlated variables. By combining the benefits of L1 and L2 regularization, the elastic net offers a versatile and effective method for regression analysis [46].

2.5.4. Dropout

Dropout is an effective regularization technique in deep learning, widely recognized for its success. Applying dropout to feed-forward neural networks yields encouraging results [47]. The function of dropout is shown in Figure 6.

Figure 6: Deep neural network (A) without applying dropout - (B) with dropout applied

3. Results

Figure 7 shows the LSTM network's joint angle estimation accuracy, based on PCC.

Figure 7: The average accuracy in estimating joint angles using the LSTM network, based on the PCC.

The PCC values, ranging from 0.73 to 0.98, indicate a high level of precision in estimating the angles of the interphalangeal joints. The highest PCC value was found for the proximal interphalangeal (PIP) joint of the ring finger, while the lowest was for the MCP joint of the thumb. To further evaluate performance, the RMSE criterion was applied, with values ranging from 5.1 to 7.5. Figure 8 shows the error values based on the RMSE criterion.

Figure 8: the average accuracy in estimating joint angles using the LSTM network, based on the RMSE.

The PIP joint of the ring finger achieved the lowest RMSE, reflecting a minimal difference between estimated and actual values. Conversely, the MCP joint of the thumb had the highest RMSE, indicating a more substantial difference between the estimated and actual values.

As shown in Figure 9, reducing the L1 regularization parameter from 0.5 to 0.0005 led to an improvement in accuracy based on the PCC metric, increasing from 86.46% to 91%. The number of epochs needed during the training process ranged from 28 to 45.

Figure 9: The impact of the LASSO (L1) regularization parameter on the accuracy of joint angle estimation with the designed LSTM network

The results of this study show that increasing the LASSO parameter value reduces the accuracy of joint angle estimation. This happens because LASSO, as a regularization technique, encourages the model to simplify by shrinking or removing certain weights from the input data. In contrast, lowering the LASSO value gives the model greater flexibility to capture complex, nonlinear relationships between muscle signals and joint angles. Lowering the LASSO value reduces the restrictions on network weights, which increases the model's complexity and requires more training epochs to reach convergence.

As illustrated in Figure 10, adjusting the L2 parameter from 0.5 to 0.0005 results in accuracy changes ranging from 58% to 96%, while the number of training epochs needed varies between 15 and 60.

Figure 10: The impact of the ridge (L2) regularization parameter on the accuracy of joint angle estimation with the designed LSTM network

The findings of the study show that a higher ridge regularization parameter leads to lower accuracy in estimating joint angles. Ridge regularization reduces large weights in the model by shrinking them; however, unlike LASSO, it does not eliminate them. In contrast, decreasing the

ridge parameter allows the model more freedom to adjust its weights, enabling it to better learn complex and detailed patterns in the data. Reducing regularization leads to a more complex model, which in turn demands more training epochs to reach convergence. As a result, achieving optimal performance requires additional training time and greater computational resources. To evaluate the impact of both the LASSO and ridge parameters in the design, the L2 parameter is set to its optimal value of 0.0005, while the L1 parameter is varied between 0.0005 and 0.5. The highest accuracy is achieved when L1 equals 0.0005. These findings are illustrated in Figure 11.

Figure 11: The influence of the Elastic Net regularization parameter on the accuracy of joint angle estimation using the developed LSTM network

Figure 12 illustrates the effect of varying the dropout rate on network performance.

Figure 12: The effect of the dropout regularization parameter on the accuracy of joint angle estimation using the developed LSTM network

As the dropout rate increases from 0.2 to 0.8, estimation accuracy declines significantly—from 0.98 to 0.63—while the required number of training epochs simultaneously decreases from 68 to 22. An increase in the dropout rate leads to a decline in the accuracy of joint angle estimation, as excessive dropout can hinder the network's ability to effectively learn and retain meaningful patterns within the data. Dropout functions as a regularization technique by randomly deactivating a portion of neurons during training, thereby preventing overfitting. According to the results of this study, the highest estimation accuracy was achieved at a dropout rate of 0.2, indicating that moderate dropout provides a suitable balance between regularization and learning capacity.

Figure 13 shows a box diagram illustrating the values of both the estimated and real finger joint angles. The LSTM-based neural network used sEMG signals to estimate joint angles. These estimates were then compared to actual joint angles measured by the CyberGlove II's bending sensors, which were accurately positioned over the finger joints. Boxplot comparisons show a strong match between the estimated and actual joint angles, especially for the PIP and distal interphalangeal (DIP) joints of the ring and little fingers. These joints exhibited minimal variance and near-identical interquartile. For most joints, there is a strong alignment between the estimated and real angle ranges, underscoring the model's robustness in capturing fine motor articulations.

Figure 13: Compare estimated and real angles of interphalangeal joints.

4. Discussions

In this study, a deep neural network model was proposed for the regression of finger- grasping movements. Due to the inherently temporal and non-stationary nature of surface EMG signals, combined with the continuous output demands of the joint angle regression task, LSTM deep neural network architecture was selected to effectively capture temporal dependencies and model complex input-output relationships. This choice was due to LSTM's ability to capture long-term dependencies in the data, thereby enhancing the accuracy of the regression task. Table 4 displays the forearm muscles involved in flexion movements [48].

Table 4: Muscles involved in finger flexion [48]

Based on the information in Table 4, the muscles involved in finger flexion for all fingers except the thumb are superficial and deep, while the thumb's flexor muscle is deep [48].

Additionally, Figure 1 shows that the electrodes are placed on the *flexor digitorum superficialis* and *flexor digitorum profundus* muscles, but not on the *flexor pollicis longus* muscle. The sEMG signals recorded from these electrodes offer detailed information on the flexion of non-thumb fingers, leading to higher prediction accuracy for non-thumb joint angles compared to those of the thumb. This conclusion is supported by the results depicted in Figures 7 and 8.

This study demonstrated that adjusting regularization settings properly is important for improving how well deep learning models estimate joint angles using surface EMG signals. Notably, applying LASSO (L1) regularization revealed that increasing the regularization strength reduced prediction accuracy. This reduction can be attributed to the sparsity-promoting nature of LASSO, which drives some model weights to zero and may inadvertently eliminate features essential for capturing the complex relationship between EMG signals and joint movement. Lowering the LASSO parameter—especially to the optimal value of 0.0005—allowed the model to preserve essential signal features, which enhanced estimation accuracy. However, this came with an increase in training time and computational load due to the added model complexity. The impact of ridge (L2) regularization also showed that applying stronger penalties limited the model's flexibility, which resulted in underfitting and a decrease in accuracy. Unlike LASSO, ridge regularization shrinks all weights uniformly without eliminating any. The best performance was obtained at a ridge coefficient of 0.0005, balancing generalization and expressiveness. Larger ridge values accelerated convergence but compromised accuracy, whereas smaller values improved precision at the cost of extended training time.

Along with using L1 and L2 penalties, the model used dropout regularization during training to reduce overfitting by randomly turning off some neurons. The optimal dropout rate was found to be 0.2, which maximized accuracy. Excessive dropout led to unstable learning and degraded performance, suggesting that moderate dropout provides a favorable trade-off between generalization and learning stability. Overall, the results show that properly adjusting regularization methods, such as LASSO, ridge, elastic net, and dropout, is key to getting the best model performance. Improper parameter selection can result in either oversimplified models with poor learning capacity or overly complex networks that are difficult to train and prone to overfitting.

From a computational complexity standpoint, regularization methods such as L1, L2, dropout, and elastic net each exhibit distinct characteristics in terms of optimization and training overhead, especially in deep learning models like LSTMs. L1 regularization promotes sparsity by penalizing the absolute values of weights. However, its non-differentiability at zero can make gradient-based optimization more challenging and may slow convergence. In contrast, L2 regularization provides a smoother and more stable optimization process due to its differentiable and convex penalty function. It introduces minimal computational overhead, making it an efficient and reliable choice. Dropout helps prevent overfitting by randomly deactivating a subset of neurons during training. While simple to implement, it introduces stochasticity into the learning process, which can increase training time and lead to higher variance in gradient updates. Elastic net combines the strengths of both L1 and L2 regularization. It encourages sparsity while maintaining stability, but achieving the right balance between the two penalties requires careful hyperparameter tuning, which can add to the computational burden. Overall, selecting the appropriate regularization technique depends on the model architecture, data characteristics, and training goals to balance performance, accuracy, and computational efficiency.

Each regularization method also carries specific side effects that influence the model's learning dynamics and generalization ability. L1 regularization can create models with very few active features, which helps in choosing important features and understanding the model better, but it might cause problems in models like LSTM that need detailed information to understand time-related patterns. L2 regularization provides smooth weight decay, helping to prevent overfitting while preserving model capacity by not eliminating features. Dropout is particularly effective in deep or over-parameterized networks, as it reduces neuron co-adaptation and improves generalization; however, if not carefully tuned, it may cause noisy training and slow convergence. Elastic net serves as a compromise between L1 and L2, offering both sparsity and regularized weight shrinkage. While this dual effect can improve robustness and generalization, especially in high-dimensional settings, it also adds complexity to model tuning. Selecting and combining these regularization strategies appropriately is essential for achieving an optimal balance between model complexity, training stability, and generalization performance. Although comparing similar works can be challenging, it is crucial for recognizing progress in a specific field. In the field of musculoskeletal modeling, He et al. created a musculoskeletal model

grounded in muscle synergy analysis, incorporating data from 32 electrodes. Their approach yielded a peak accuracy of 0.92, representing the highest performance reported in their research [9]. Kim et al. developed a combined muscle synergy and musculoskeletal model to estimate wrist joint angles, achieving a maximum accuracy of 84% [10]. Using fourteen surface electrodes integrated with a Kalman filter algorithm, Zhang et al. estimated finger kinematics with a prediction accuracy of 0.73 across six finger joint angles [11]. Gilstrep et al. also employed a musculoskeletal modeling approach to continuously estimate index finger movements, reporting an RMSE of 2.15 [12]. However, this method necessitates the experimental calibration of multiple biomechanical parameters, which presents major challenges for practical application and limits its overall accuracy. Some studies have worked on decomposing sEMG signals into individual motor unit action potentials (MUAPs) to better understand muscle activation in a more meaningful way. Roy et al. used sEMG signals from finger muscles with 128 electrodes and achieved 94% accuracy in predicting joint angles [13]. Dai et al. conducted a study in which they estimated MCP joint extension angles using 160 electrodes. Their approach followed a two-stage process: first, classifying finger movements, and then applying second-order polynomial regression to predict joint angles. The results demonstrated that their method achieved an estimation accuracy exceeding 0.8 [49]. By analyzing motor unit discharge timings from high-density EMG recordings, Chen et al. achieved 85% accuracy in predicting finger joint kinematics [49]. However, despite its theoretical advantages, this method presents substantial practical limitations. The decomposition process is computationally intensive and requires sophisticated algorithms to separate overlapping MUAPs, especially during dynamic contractions. High-density electrode arrays complicate the experimental setup and introduce additional challenges, including the need for thorough skin preparation, accurate electrode placement, increased risk of signal crosstalk, and reduced user comfort, particularly in real-time or wearable applications.

The second method, based on conventional machine learning techniques, such as Zhang et al., reached an accuracy of 0.91 in predicting MCP joint angles. Their study used six electrodes and applied Gaussian process regression to link sEMG features with hand movements [15]. However, these methods frequently require intricate and time-consuming feature extraction procedures, which can lead to the loss of important signal information.

The third category involves the application of traditional neural networks. Batayneh et al. created a neural network to connect sEMG signals with hand joint angles and achieved 79.8% accuracy [19]. Gao et al. implemented a control strategy using an NARX, which effectively estimated joint angles by analyzing sEMG signals. In this study, principal component analysis (PCA) and kalman filtering were also utilized, which increased the computational complexity and processing time and may have led to the loss of valuable signal information [20]. Tacca et al. employed a traditional neural network and 150 electrodes to estimate joint angles during four specific hand movements, reporting an accuracy of 0.88 [21]. However, this method heavily relies on the incorporation of numerous parameters and the presence of high-quality data.

Deep convolutional neural networks have also been used for joint angle estimation. Qin et al. utilized a convolutional neural network to estimate the wrist joint angle, achieving an accuracy of 89% in their study [26]. Guo et al. used a convolutional model to estimate finger movements and achieved their highest accuracy with a correlation coefficient of 0.82 [27]. Sun et al. developed a multi-stream convolutional neural network for continuous finger gesture recognition, attaining an accuracy of 0.84 [30]. An et al. predicted finger joint angles during grasping movements using a Convolution Layers Transformer, achieving an average accuracy of 0.84 [33]. Although CNNs are effective at extracting complex features from two-dimensional data, using them alone for EMG signal analysis has some limitations. EMG signals have a time-based, sequential structure, which requires models that can capture changes and patterns over time. Since CNNs are not specifically designed to handle temporal information, they may not perform as well in estimating joint angles.

Chen et al. employed an LSTM network to predict shoulder joint angles. The model achieved a prediction accuracy of 0.91 for touch tasks and 0.81 for compound tasks [24]. Tang et al. used an LSTM network to predict elbow and wrist movements, achieving an RMSE of 0.1025 [28]. Guo et al. predicted finger kinematics using a multi-attention feature fusion network, attaining an accuracy of 84% [31]. Wang et al. utilized a hybrid unsupervised domain adaptation network to estimate joint angles for application in myoelectric control systems, resulting in an estimation accuracy of 87% [32]. Wang et al. introduced an ensemble learning method for estimating MCP joint angles during finger movements, achieving an estimation accuracy of 89.7% [34]. Lin et al. created a method to predict ten finger joint angles during eight grasping movements using the correlation coefficient criterion, achieving an average accuracy of 0.82. Their approach incorporated the efficient multiple self-attention model [51]. Anam et al. adopted a two-stage approach to estimate hand joint kinematics, beginning with dimensionality reduction of sEMG signals, followed by the application of an LSTM network. Their method yielded an estimation accuracy of 93% [36]. Table 5 provides a summary comparing the current research with related previous studies.

Table 5: Notable studies on joint angle estimation

Some of the studies above focused on decomposing EMG signals into motor units [13, 49, and 50], which, while valuable, required a large number of electrodes and involved complex, time-consuming procedures. Additionally, several earlier works employed traditional machine learning techniques [11, 15]; however, these methods were not only costly and time-intensive but also posed the risk of losing critical signal information. Moreover, some of these studies limited their analysis to the kinematics of a single joint or a small number of joints [9, 11, 12, 15, 34, and 51], thus restricting the generalizability of their findings. Some studies have utilized CNNs [26, 27, 30, and 33]. While CNNs excel at extracting complex features from two-dimensional data, depending solely on them for analyzing EMG signals presents certain limitations, especially in capturing the temporal dynamics that are inherent in EMG patterns. Several previous studies

have incorporated pre-processing and post-processing techniques for sEMG signal analysis and joint angle estimation [25, 36]. While these methods—such as filtering, normalization, feature extraction, and dimensionality reduction—can enhance signal quality and reduce noise, they often come at the cost of increased computational complexity and longer processing times. In particular, techniques like PCA or ICA may discard components deemed less significant, potentially eliminating subtle but physiologically relevant information embedded in the raw sEMG signals. This loss of useful signal features can ultimately impair the performance of prediction models, especially in real-time or resource-constrained applications where both accuracy and efficiency are critical.

This study demonstrates that meticulous tuning of deep network parameters can greatly improve model performance. Using the Ninapro DB4 dataset, the proposed model achieves an accuracy of up to 98% (mean: 96%) in estimating joint angles across multiple subjects. The research proves that accurate joint angle estimation can be achieved with only 10 sEMG electrodes for all finger joints. This efficiency highlights the model's practicality and suitability for real-time use in clinical settings. A muscle-level analysis was conducted to interpret variations in accuracy across joints, providing insights into the EMG-muscle-joint relationship during finger flexion. These findings offer valuable biomechanical understanding and guidance for optimizing electrode placement in future designs. The approach does not rely on complex pre-processing or post-processing of the signals, making it a promising candidate for real-time applications.

Since this study investigates the impact of regularization parameters, the number of training epochs required varied accordingly, ranging from a minimum of 25 to a maximum of 68, depending on the parameter values. A CPU Intel Core i7-2.70 GHz was used to perform all implementations. The time and implementation analysis conducted in this study, along with that of several related works that have reported these aspects, is presented in Table 6.

Table 6: Comparison of the implementation and time analysis between the proposed model and related researches

5. Conclusions

This study presents promising results in the estimation of finger joint angles. Given the sequential and stochastic nature of sEMG signals, deep LSTM neural networks proved to be highly effective for modeling such data. The findings also show that precise hyperparameter tuning can substantially enhance model performance in joint angle estimation tasks. The proposed LSTM-based model, evaluated on the Ninapro DB4 dataset, achieved a peak accuracy of 98% and an average accuracy of 96% across multiple subjects, highlighting its robustness and strong ability to generalize across individuals. Regularization proved to be a key factor in reaching this level of performance. Specifically, setting the LASSO (L1) regularization coefficient to 0.0005 helped the model preserve essential signal features, resulting in more

accurate predictions. Similarly, setting the ridge (L2) coefficient to 0.0005 provided a well-balanced compromise between the model's generalization ability and its representational capacity. While higher L2 values accelerated convergence, they tended to reduce accuracy; conversely, lower values slightly improved precision but extended training time. Furthermore, a dropout rate of 0.2 was found to be optimal, efficiently mitigating overfitting while maintaining high overall accuracy. These results highlight the critical role of systematic hyperparameter optimization in developing high-performing models for biomechanical signal processing and analysis. A detailed physiological analysis was also conducted to explain the variability in the accuracy of interphalangeal joint angle estimation and to identify the muscles contributing.

Finger joint angle estimation will be performed across a broader range of movement types, including fine-grained and high-speed motions. To this end, various LSTM variants such as bidirectional LSTM (BiLSTM), Gated Recurrent Units (GRUs), and Layer-Normalized LSTMs will be evaluated to determine their capacity for capturing long-term dependencies and complex temporal dynamics in sequential sensor data. Furthermore, the integration of multi-branch or hierarchical architectures will be explored, allowing the model to process multi-modal signals (e.g., accelerometer, gyroscope, and magnetometer data) in parallel before temporal fusion. Attention mechanisms, both temporal and spatial, will be incorporated to enable the network to selectively focus on the most informative time steps and sensor channels, which is particularly beneficial for noisy or redundant input data.

Acknowledgments

The authors declare that no funds, grant, or other support were received during the preparation of this manuscript.

Funding: The authors declare that no funds, grant, or other support were received during the preparation of this manuscript.

Competing Interests: The authors have no relevant financial or non-financial interests to disclose.

Data availability statement: The datasets analysed during the current study are DB4 Ninapro and available in http://ninaweb. hevs.ch/

Ethics approval

We used public Nina-pro Dataset. http:// ninaweb.hevs.ch/

Consent to participate

We used public Nina-pro Dataset. http:// ninaweb.hevs.ch/

Consent to publish

We used public Nina-pro Dataset. http:// ninaweb.hevs.ch/

References

1 Feng, J., Yang, M. J., Kyeong, S., et al. "Hand grasp motion intention recognition based on high-density electromyography in chronic stroke patients", *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. Annual International Conference, 1–4.* (2023). https://doi.org/10.1109/EMBC40787.2023.10 340346

2 Li, Y., Lian, Y., Chen, X., et al. “Effect of task-oriented training assisted by force feedback hand rehabilitation robot on finger grasping function in stroke patients with hemiplegia: a randomized controlled trial”, *Journal of Neuro Engineering and Rehabilitation*, 21, 77 (2024). https://doi. org/10.1186/s12984-024-01372-3
3 Shen, S., Gu, K., Chen, X.R, et al. “Movements classification of multi-channel sEMG based on CNN and stacking ensemble learning”, *IEEE Access*. 7:137489-137500 (2019). https://doi.org/10.1109/ACCESS.20 19. 2941977.
4 Kahn, X., Yang, D., Cao, L., et al. “A novel PSO-based optimized light weight convolution neural network for movements recognizing from multichannel surface electromyogram”, *Complexity*, 6642463 15 (2020). https://doi. org/10.1155/2020/6642463.
5 Asif, A.R, Waris, A., Gilani, S.O., et al. “Performance evaluation of convolutional neural network for hand gesture recognition using EMG”, *Sensors*. 20, 6, 1642 (2020). https://doi.org/10.3390/s20061642
6 Chen, X., Li, Y., Hu, R., et al. “Hand gesture recognition based on surface electromyography using convolutional neural network with transfer learning method”, *IEEE Journal of Biomedical and Health Informatics*. 25, 4: 1292-1304 (2021). https://doi.org/10.1109/ JBHI.20203009383.
7 Nasri, N., Orts-Escolano, S., Gomez-Donoso, F. “Inferring static hand poses from a low-cost non-intrusive sEMG sensor”, *Sensors*. 19(2):371. (2019). https://doi.org/10.3390/ s19020371.
8 Simão, M., Neto, P., Gibaru, O., “EMG-based online classification of gestures with recurrent neural networks”, *Pattern Recognition Letters*, 128 (128), 45-51, (2019). https://doi.org/10. 1016/j.patrec.2019.07.021.
9 He, Z., Qin, Z., Koike, Y. “Continuous estimation of finger and wrist joint angles using a muscle synergy based musculoskeletal model”, *Appl. Sci*, 12(8):3772 (2022). https:// doi.org/10.3390/app12083772.
10 Kim, Y., Staporncha isit, S., Kambara, et al. “Muscle synergy and musculoskeletal model-based continuous multi-dimensional estimation of wrist and hand motions”, *J. Healthc. Eng*., 5451219 (2020). https://doi.org/10.1155/2020/ 5451219.
11 Zhang, H., Peng, B., Tian, L., et al. “Continuous kalman estimation method for finger kinematics tracking from surface electromyography”, *Cyborg Bionic Syst.* 5:0094 (2024). https://doi.org/10.34133/cbsy stems.0094.
12 Gilstrap, A., Phan, T., M. Alghamdi, T., et al. “EMG-based continuous estimation of index finger movements with varying inter joint coordination patterns by modeling musculoskeletal dynamics”, *IEEE Access* 99:1-1 (2025) https://doi.org/10.1109/ACCESS.20 25.3528902.
13 Roy, R., Zheng, Y., D. G. Kamper, D. G., et al. “Concurrent and continuous prediction of finger kinetics and kinematics via motor neuron activities”, *IEEE Transactions on Biomedical Engineering*, 70(6): pp. 1911-1920 (2023). https ://doi.org/10.1109/TBME.2022. 3232067.
14 Ma, X., Liu, Y. “Continuous estimation of knee joint angle based on surface electromyography using a long short term memory neural network and time- advanced feature”, *Sensors*, 20, 17 (2020). https://doi. org/10.3390/s2017496 6.
15 Zhang, Q., Pi, T., Liu, R., et al. “Simultaneous and proportional estimation of multijoint kinematics from EMG signals for myo control of robotic hands”, *IEEE/ASME Trans. Mechatronics*, 25(4): 1953-60 (2020). https:// doi.org/10.1109/ tmech.2020.2999532.
16 Gao, Y., Luo, Y., Zhao, J., et al. “sEMG-angle estimation using feature engineering techniques for least square support vector machine”, *J. Technol. Health Care*, 27, s1, pp. 31-46 (2019). https://doi.org/10.3233/thc-199005.
17 Hu, Y., Wong, Y., Wei, W., et al. “Novel attention-based hybrid CNN-RNN architecture for sEMG-based gesture recognition”, *Plus One*. 13(10) (2018). https://doi.org/10.1371/ journal. pone.0206049.
18 Tufail, S., Riggs, H., Tariq, M., et al. “Advancements and challenges in machine learning: a comprehensive review of models, libraries, applications, and algorithms”, *Electronics*, 12, 1789 (2023). https://doi.org/ 10.3390/electronics12081789.
19 Batayneh, W., Abdulhay, E., Alothman, M. “Prediction of the performance of artificial neural networks in mapping sEMG to finger joint angles via signal pre-investigation techniques”, *Heliyon*. 6(4) (2020). https:// doi.org/10.1016/j.heliyon.2020.e03669.
20 Gao, Z., Tang, R., Huang, Q., et al. “A multi-DoF prosthetic hand finger joint controller for wearable sEMG sensors by nonlinear autoregressive exogenous model”, *Sensors*, 21(8), 2576 (2021). https://doi.org/10.3390/s21 082576.
21 Tacca, N., Dunlap, C., Donegan, S.P. et al. “Wearable high-density EMG sleeve for complex hand gesture classification and continuous joint angle estimation”, *Scientific Report* 14, 18564 (2024). https://doi.org/ 10.1038/s41598-024-64458-x

22 Rahmani, M., Kazemi, A., Jalili Aziz, M., "A systematic review on medical image segmentation using deep learning", *Scientia Iranica*, (2024). https://doi.org/10.24200/sci.20 24.61686.7441.
23 S. Jameyan, M., A. Pajouh, M.A., Moradi, M.H., "Deep LSTM neural networks in kinematic estimation of the finger interphalangeal joint's angles using surface electromyogram signals", *Scientia Iranica*, (2025). https://doi.org/10.24200/sci.2025.653 19.9419.
24 Chen, Y., Yu, S., Ma, K., et al. "A continuous estimation model of upper limb joint angles by using surface electromyography and deep learning method", *IEEE Access*. 7:174940-50 (2019). https://doi.org/10.1109/access.2019.29 56951.
25 Kim, Y., Stapornchaisit, S., Miyakoshi, M., et al. "The effect of ICA and non-negative matrix factorization analysis for EMG signals recorded from multi-channel EMG sensors", *Front. Neurosci*. 14:600804 (2020). https:// doi.org/10.3389/fnins.2020.600804.
26 Qin, Z., Stapornchaisit, S., He, Z., et al. 2021. "Multi joint angles estimation of forearm motion using a regression model", *Front. Neurorobot*. 15: 685961 (2021). https://doi.org/ 10.3389/fnbot.2021.685961.
27 Guo, W., Ma, C., Wang, Z., et al. "Long exposure convolutional memory network for accurate estimation of finger kinematics from surface electromyographic signals", *J Neural Eng*, 3(18) (2021). https://doi.org/10.1088/1741-2552/abd46 1.
28 Tang, G., Sheng, J., D. Wang, D, et al. "Continuous estimation of human upper limb joint angles by using PSO-LSTM model", *IEEE Access*. 9: 17986-17997 (2021). https:// doi.org/10.1109/ACCESS.2020.30478 28.
29 Wen, Y., Kim, S.J., Avrillon, S., et al. "A deep CNN framework for neural drive estimation from HD-EMG across contraction intensities and joint angles", *IEEE Transactions Neural System Rehabilitation Engineering*; 30, pp. 2950-59 (2022). https://doi.org/10.1109/ tnsre.2022.3215246.
30 Sun, X., Liu, Y., Niu, H. "Continuous gesture recognition and force estimation using sEMG signal", *IEEE Access*, vol. 11, pp. 118024-118036 (2023). https://doi.org/10.1109/ACC ESS.2023.3323586
31 Guo, W., Jiang, N., Farina, D., et al., "Multi-attention feature fusion network for accurate estimation of finger kinematics from surface electromyographic signals", *IEEE Transactions on Human-Machine Systems*, 53(3), 512-519 (2023). https://doi.org/10.1109/ THMS.2023. 3252817.
32 Wang, L., Li, X., Chen, Z., et al., "A novel hybrid unsupervised domain adaptation method for cross-subject joint angle estimation from surface electromyography", *IEEE Robotics and Automation Letters*, 8(11), 7257-7264 (2023). https://doi.org/10.1109/LRA.2023. 3317680.
33 An, Z., et al., "SCTNET: Shifted windows and convolution layers transformer for continuous angle estimation of finger joints using sEMG", *IEEE Sensors Journal*, 24(16), 27007-27016, 15 (2024). https://doi.org/10.1109/JSEN. 2024. 3423795.
34 Wang, H., Tao, Q., Zhang, X. "Ensemble learning method for the continuous decoding of hand joint angles", *Sensors* 24,660 (2024). https://doi.org/10.3390/s24020660.
35 Lin, C., Zhang, X. "Fusion inception and transformer network for continuous estimation of finger kinematics from surface electromyo graphy". *Front. Neurorobot*. 18:1305605 (2024) https://doi.org/10.3389/fnbot.2024.130 5605.
36 Anam, K., Rizal, N.A., Ilyas, Z. et al. "Evaluation of long short term memory on simultaneous and proportional myoelectric control system for individual finger movements", *Res. Biomed. Eng*. 41, 9 (2025). https://doi.org/10.1007/s42600-024-00382-3.
37 Pizzolato, S., Tagliapietra, L., Cognolato, M., et al. "Comparison of six electromyography acquisition setups on hand movement classification tasks", *Plos one*, 12(10), e0186132 (2017). https://doi.org/10.1371/journal.pone.0186132
38 Atzori, M., Gijsberts, A. "Characterization of a benchmark database for myoelectric movement classification", IEEE *Trans. Neural Syst. Rehabil*. 23(1):73-83. (2013). https://doi.org/ 10.1109/tnsre.2014.2328 495.
39 Jarque-Bou, N.J, Atzori, M., Müller, H. "A large calibrated database of hand movements and grasps kinematics", *Sci. Data*. 7(12) (2020). https://doi.org/10.1038/s41597-019-0349-2
40 Hochreiter, S., Schmidhuber, J. "Long short-term memory", *Neural comput*. 9(8):1735-1780. (1997). https://doi.org/10.1162/neco.1 997.9.8.1735.
41 Greff, K., Srivastava, R.K. "LSTM: A search space odyssey", *IEEE Trans. Neural Networks and Learning Syst*. 28(10):2222-32 (2016). https://doi.org/10.1109/tnnls.2016. 2582924.
42 Zhang, Z., Li, B., Zhang, J. "An improved BGE-Adam optimization algorithm based on entropy weighting and adaptive gradient strategy", *Symmetry*, 16(5), 623.(2024) https:// doi.org/10.3390/sym16050623
43 Yokochi, C., Bispo, R., Ricardo, F., et al. "Regularization methods for high-dimensional data as a tool for sea food traceability", *J Stat Theory Pract*. 17, 44. (2023). https://doi.org/ 10.1007/s42519-023-00341-8.

44 Pak, A., Rad, A.K., Nematollahi, M.J. et al. “Application of the Lasso regularization technique in mitigating overfitting in air quality prediction models”. *Sci Rep* 15, 547 (2025). https://doi.org/10.1038/s41598-024-84342-y.
45 Hastie, T. “Ridge regularization: An essential concept in data science”, *Techno metrics*, *62*(4), 426–433. (2020). https://doi.org/10. 1080/0040 1706.2020.1791959.
46 Jiao, J., Su, K.“A new sigma-pi-sigma neural network based on L1 and L2 regularization and applications”, *AIMS Mathematics* 9, (3), 5995-6012. (2024). https://doi.org/10.3934/math.202 4293.
47 Tian, Y., Zhang, H., & Zhang, Z. “Dropout with expectation-linear regularization for robust learning”. *Neurocomputing*, 501, 744–755 (2022). https://doi.org/10.1016/j.neucom. 2021.08.107
48 Tuncay, S., Ebraheim, N. A. “The forearm: anatomy of muscle compartments and nerves”. *American Journal of Roentgen ology*, 174(1), 151–159 (2000). https://doi.org /10.2214/ajr. 174.1.1740151.
49 Dai, C., Hu, X. “Finger joint angle estimation based on motor neuron discharge activities”, *IEEE J. Biomed. Heal. Informatics*. 24(3): 760-767, (2020). https://doi.org/10.1109/jbhi.2019. 2926 307.
50 Chen, D., Zhu, X. “Prediction of finger kinematics from discharge timings of motor units: Implications for intuitive control of myoelectric prostheses”, *J. of Neural Eng*, 16, 2 (2019). https://doi.org/10.1088/1741-2552/aaf4c 3.
51 Lin, C., Zhao, C., Zhang, J., et al. “Continuous estimation of hand kinematics from electromyographic signals based on power and time-efficient transformer deep learning network”, *33 IEEE Transaction on neural systems and rehabilitation engineering* (2025). https://doi.org/10.1109/TNSRE.2024.3514938

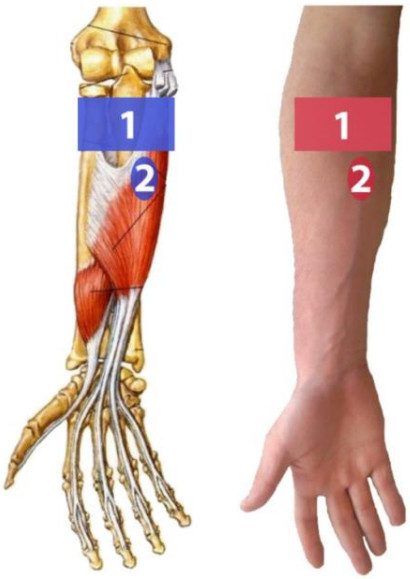


Figure 1

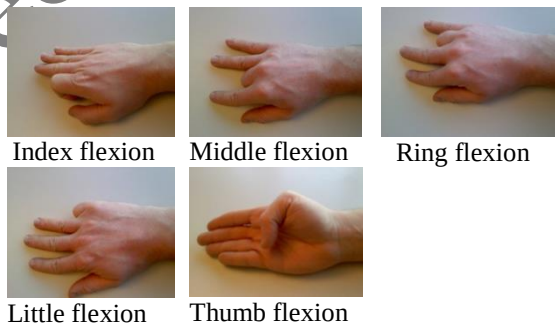


Figure 2

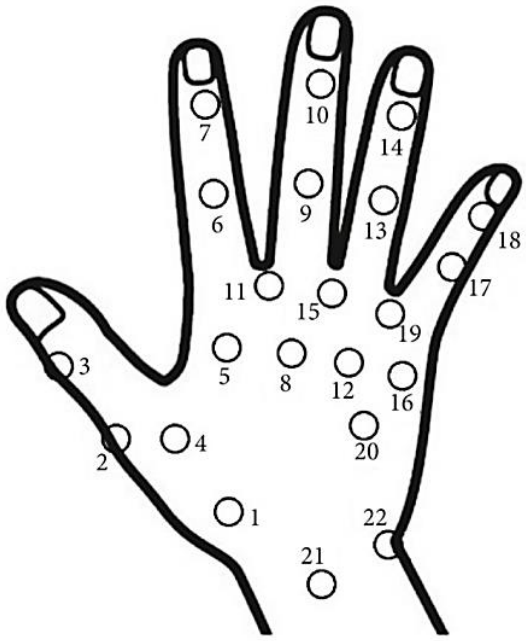


Figure 3

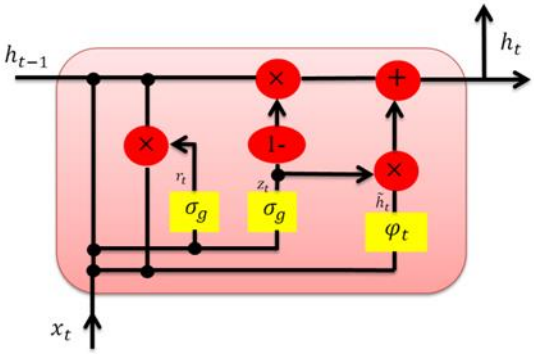


Figure 4

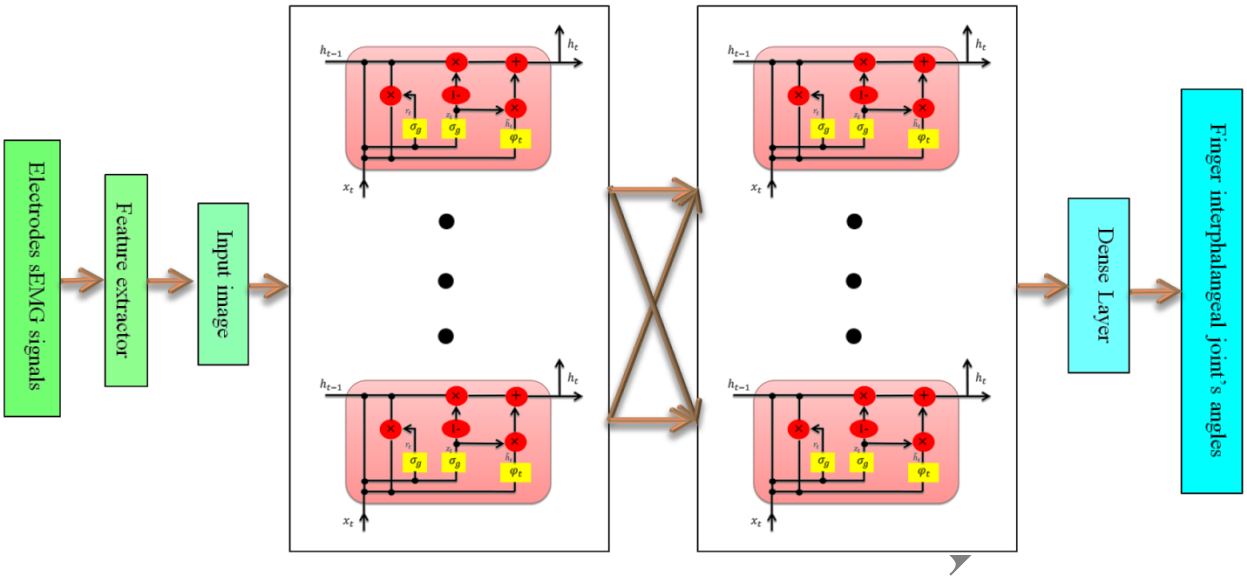


Figure 5

(A)

(B)

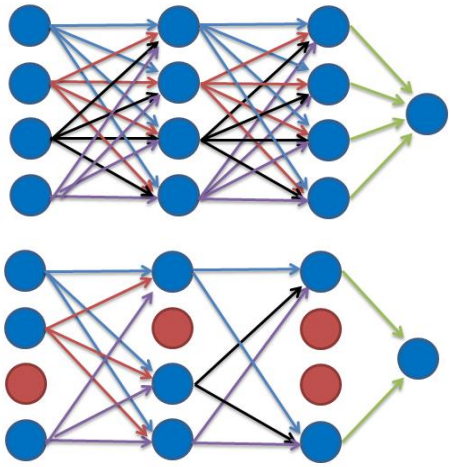


Figure 6

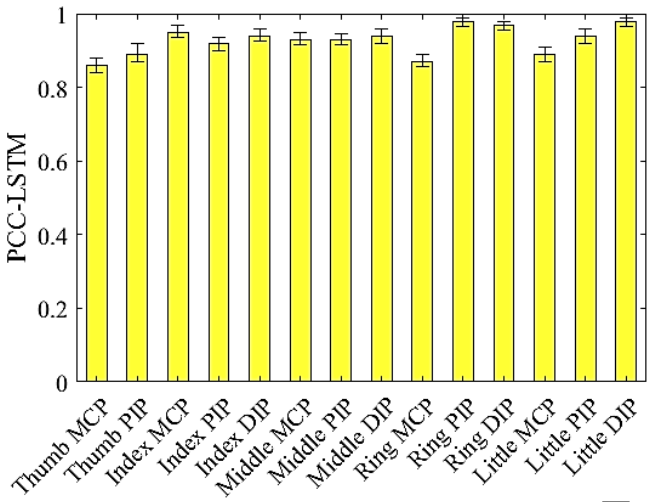


Figure 7

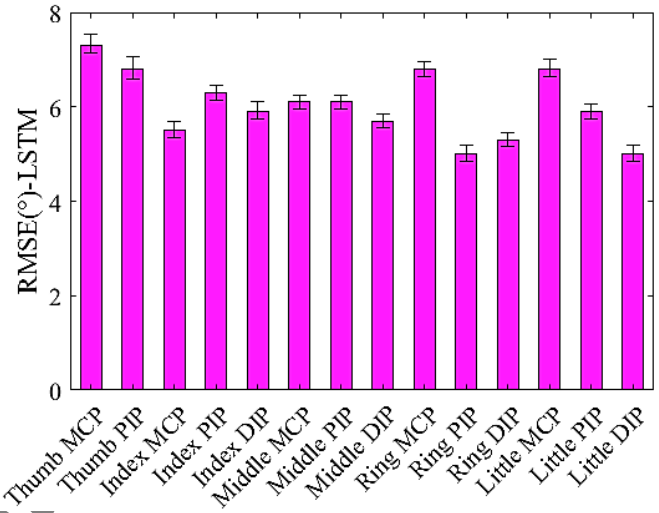


Figure 8

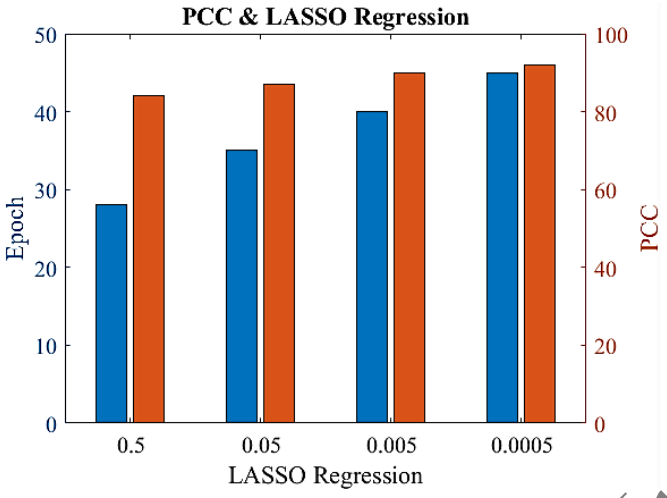


Figure 9

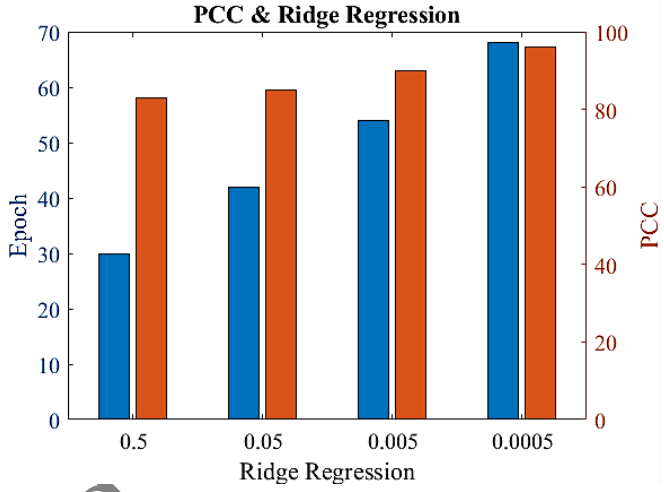


Figure 10

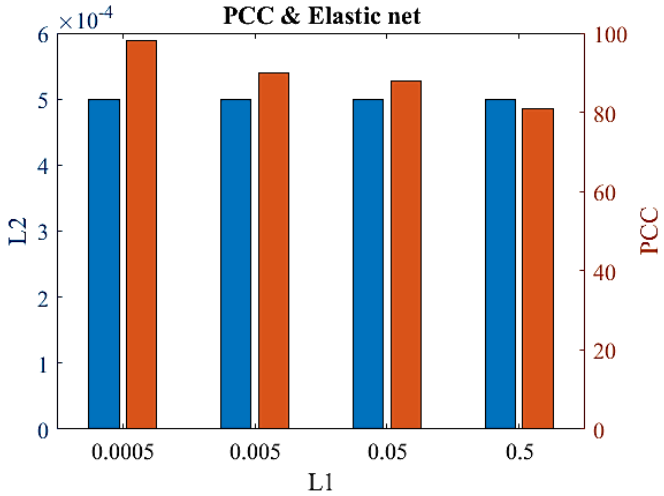


Figure 11

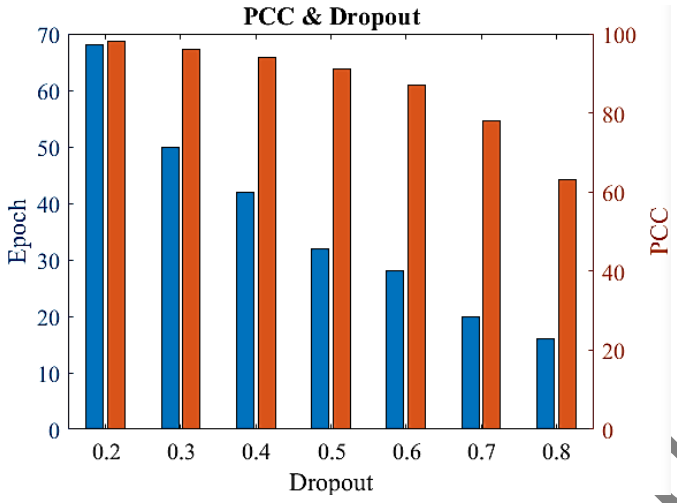
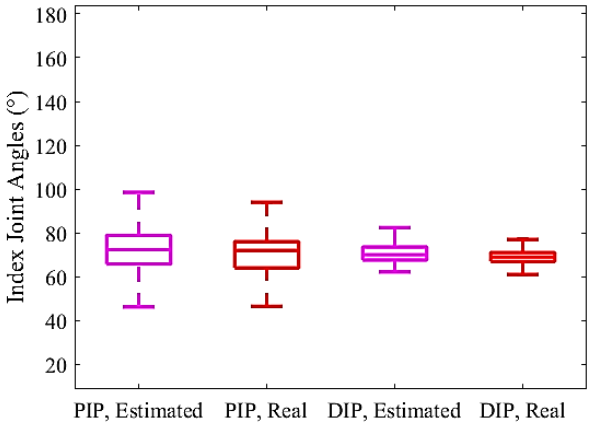
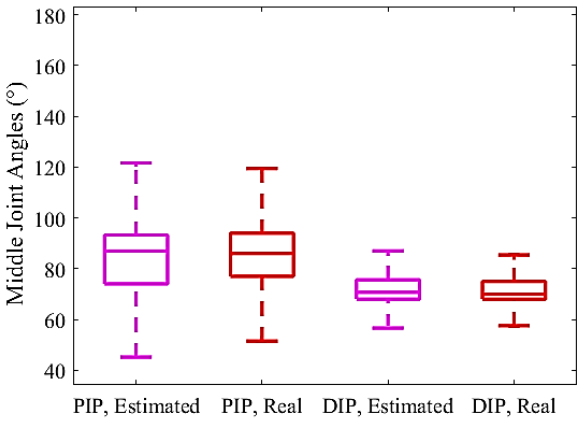


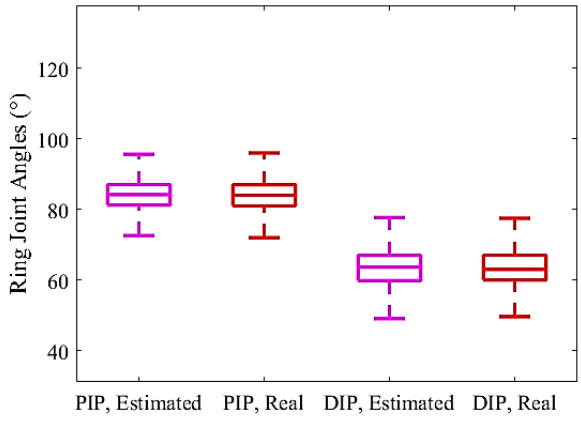
Figure 12



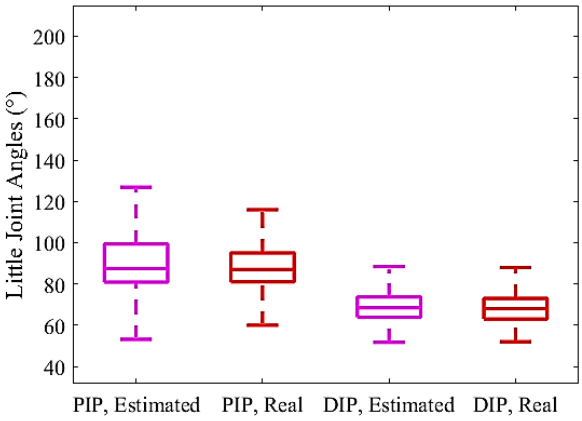
(A) Index joint angles



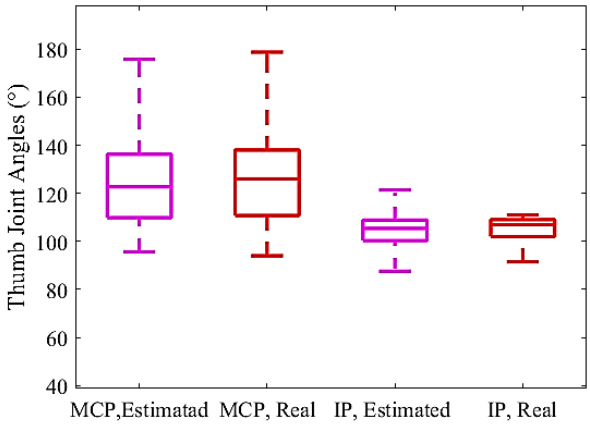
(B) Middle joint angles



(C) Ring joint angles



(D) Little joint angles



(E) Thumb joint angles

Figure 13

Table 1

LSTM vectors & parameters	Description
W, U	Weight matrices
b	Bias vector
f_t	Forget gate's activation vector
σ_g	Sigmoid function
σ_c, σ_h	Hyperbolic tangent function
x_t	Input vector to the LSTM unit
h_t	Hidden state vector
i_t	Input/update gate's activation vector
o_t	Output gate' activation vector
c_t	Cell state vector
y	Block output

Table 2

Parameter	Description
y_{pred}	Estimated angles
y_{true}	Actual angles
$\overline{y}_{pred}$	Mean of Estimated angles
$\overline{y}_{true}$	Mean of Actual angles

Table 3

Parameter	Description
$Loss$	The prediction error
λ	regularization hyperparameter
m	the number of training examples used to normalize the cost function
w	model weights

Table 4

Muscle	Task- Placement
flexor digitorum superficialis	Fingers flexion- Superficial
flexor digitorum profundus	Hand / Joints flexion- Deep
flexor pollicis longus	Thumb flexion- Deep

Table 5

Research	Implementation method	Accuracy
He et al. [9]	Muscle synergy model- MCP joint angles for the thumb, index, and middle	92%
Kim et al. [10]	Muscle synergy- Wrist	84%
Zhang et al. [11]	SPGP - Finger MCP joints	91%
Gilstrep et al. [12]	Musculoskeletal model- Index joint	RMSE:2.15
Roy et al.[13]	Decompose sEMG to MUAPs- Joint angle	94%
Dai et al. [49]	Decompose sEMG to MUAPs -MCP joint angle	80%
Chen et al. [50]	By analyzing motor unit discharge timings	85%
Zhang et al. [15]	Kalman-Six finger joint	73%
Batayneh et al. [19]	RBFNN- Finger joint angle	79.8%
Tacca et al. [21]	Neural network-Finger joint angle	88%
Chen et al. [24]	LSTM - Shoulder joint angle	81%-compound task
Qin et al. [26]	CNN- Wrist joint angle- Wrist flexion & extension	89%
Guo et al. [27]	CNN-Finger kinematic	82%
Tang et al. [28]	LSTM- Elbow joint and wrist joint	RMSE:0.1025
Sun et al. [30]	CNN-Continues recognition finger	84%
Guo et al. [31]	Multi attention feature fusion- Finger kinematic	84%
Wang et al. [32]	Hybrid unsupervised domain adaptation network	87%
An et al. [33]	CNN- Finger joint angle	84.35%
Wang et al. [34]	Ensemble learning- MCP joint angles	89.7%
Lin et al. [35]	Inception-Transformer network- Ten finger joint	81.24%
Lin et al. [51]	Encoder-Decoder-Ten finger joint	82%
Anam et al. [36]	ICA-LSTM- Hand joint kinematics	93%
This study	LSTM-Thumb, Index, Middle, Ring, and Little joint angel	96% (Mean)- 98% (Max)

Table 6

Research	Training time(epoch)	Implementation
Gilstrep [12]	200	Intel Core i7-2.80GHz
Batayneh 19]	1000	4-core, 2.8 Hz
Tacca [21]	200	Intel Core i7-2.50GHz
Sun [30]	200	Intel Core i7-3.20GHz
Lin [35]	100	NVIDIA GeForce RTX 3060 GPU
Lin[38]	1000	Intel i5-300HQ
This study	Min: 25-max 68	Intel Core i7-2.70GHz

Masoud Saheb Jameyan, Ph.D. Candidate
Biomedical Eng. Department,
Amirkabir University of Tech,
424 Hafez Ave.,
Tehran, Iran.
Postal Box: 15875-4413
Cell: (+98) 912-8063375
Email: sjameyan@aut.ac.ir

Mohammad Ali Ahmadi-Pajouh, Ph.D.
Biomedical Eng. Department,
Amirkabir University of Tech,
424 Hafez Ave.,
Tehran, Iran.
Postal Box: 15875-4413
Cell: (+98) 912-6304792
Work: (+98)21-64545570,
Email: pajouh@aut.ac.ir

Mohammad Hassan Moradi, Ph.D.
Biomedical Eng. Department,
Amirkabir University of Tech,
424 Hafez Ave.,
Tehran, Iran.
Postal Box: 15875-4413
Work: (+98)21-64542399,
Email: mhmoradi@aut.ac.ir

Biography

Masoud Saheb Jameyan

Masoud Saheb Jameyan obtained his bachelor's and master's degrees in Control Electrical Engineering and Electronics Electrical Engineering from Isfahan University of Technology and Noshirvani University of Technology, respectively. He is currently a Ph.D. candidate in Biomedical Engineering (Bioelectric) at Amirkabir University of Technology. His Ph.D. research focuses on the classification and regression of human hand and finger movements using electromyogram signals, machine learning, and deep learning. He has expertise in designing digital electronic boards, machine learning, and deep learning.

Mohammad Ali Ahmadi-Pajouh

Dr. Mohammad Ali Ahmadi-Pajouh is an accomplished researcher and academic specializing in biomedical engineering, with a focus on human motion analysis and cognitive modeling. He earned his Ph.D. in Biomedical Engineering from Amirkabir University of Technology, Tehran, Iran, in 2012. His doctoral research centered on estimating human arm stiffness using kinematic and electromyogram signals, highlighting his expertise in integrating biomechanics and signal processing. During his Ph.D., Dr. Ahmadi-Pajouh was also a visiting scholar at Johns Hopkins University, USA, where he explored the context-dependent modulation of sensory feedback gains. His earlier academic achievements include a Master's degree in Biomedical Engineering from Amirkabir University and a Bachelor's degree in Electrical Engineering (Control Engineering) from K.N. Toosi University of Technology.
Since 2013, Dr. Ahmadi-Pajouh has served as an Assistant Professor in the Biomedical Engineering Department at Amirkabir University of Technology. Additionally, he has contributed to interdisciplinary research as a faculty member of the Independent Mechatronics Group and previously at the Iranian Research Institute for Cognitive

Sciences. His work spans cognitive modeling, human path planning, and sensory-motor control, combining his engineering background with innovative approaches to understanding human movement and behavior. Dr. Ahmadi-Pajouh's ongoing research reflects his commitment to advancing biomedical engineering and cognitive science.

Mohammad Hassan Moradi

Mohammad Hassan Moradi received his B.Sc. and M.Sc. degrees in Electronic Engineering from University of Tehran, Iran, in 1988 and 1990, respectively. He received his PhD degree from Tarbiat Modarres University, Tehran, Iran, in 1995. He has been with the faculty of biomedical engineering, Amirkabir University of Technology, Tehran, Iran, since 1995. His primary research and teaching interests include design, manufacturing and application of medical instrumentation, biomedical signal processing, cognitive neuroscience, neuro-marketing, wavelet system design, time-frequency transforms and deep fuzzy neural systems. He has published over 200 technical papers in various high impact factor peer-reviewed journals. Moreover, he presented more than 400 papers in international conferences. Professor Moradi translated a book on wavelet signal processing into Persian. He has written 20 book chapters as well.