# A Deep Spatio-Temporal Model for Decoding Simultaneous and Continuous Hand Movements from Surface Electromyography

Sukrit Ghosh, Yashkrit Singh, Krishan K Sah, Garima Bhandari, *Member, IEEE RAS*, D.S.V Bandara, *Member, IEEE*, and Jyotindra Narayan, *Member, IEEE*

***Abstract*—Predicting continuous finger kinematics from surface electromyography (sEMG) signals provides crucial input for the intuitive proportional control of extreme upper-limb robotic prostheses. However, this regression task remains a complex challenge due to the noisy and non-linear dynamics of muscle activations. In this study, a hybrid deep spatio-temporal model is proposed that combines a Convolutional Neural Network (CNN) and a Long Short-Term Memory (LSTM) network to predict simultaneous finger joint angles from raw sEMG signals. Evaluated on 27 subjects from the Ninapro DB1 dataset using a rigorous repetition-wise split, the model was tested across three input window sizes (400 ms, 600 ms, and 800 ms). Results demonstrate that the Hybrid CNN-LSTM consistently outperforms standard LSTM and Bidirectional LSTM (BiLSTM) baselines. The optimal 800 ms window achieved the best overall performance ($R^2 = 0.7733$, $PCC = 0.8841$, $RMSE = 11.65°$), demonstrating high-fidelity tracking particularly in the highly active digits. These findings highlight the feasibility of deploying efficient, single-modality deep learning estimators for robust, real-time prosthetic control without requiring complex multimodal sensor fusion.**

***Index Terms*—surface electromyography, continuous motion estimation, proportional control, CNN-LSTM, Ninapro dataset, finger kinematics**

## I. INTRODUCTION

Surface electromyography (sEMG) provides a non-invasive interface by recording muscle electrical activity, which occurs 100–300 ms prior to actual mechanical movement [1]. This predictive capability is vital for developing intuitive prosthetic control systems [2], [3]. Historically, prosthetic systems relied on discrete gesture classification, restricting users to predefined poses [4], [5]. To enable flexible, natural control for stroke survivors or individuals with neuromuscular disorders, mapping sEMG signals continuously to joint angles, a regression scheme, has emerged as a superior alternative [6]. Unlike classification, continuous regression allows simultaneous proportional control of independent finger joints, mimicking human dexterity [7].

Sukrit Ghosh, Yashkrit Singh, Krishan K Sah, and Jyotindra Narayan are with the Department of Mechanical Engineering, Indian Institute of Technology (IIT) Patna, Bihar, India, 2201me69_sukrit@iitp.ac.in, 2201me76_yashkrit@iitp.ac.in, krishan_2521me07@iitp.ac.in, jnarayan@iitp.ac.in

Garima Bhandari is with Department of Mechanical Engineering, California State University, Sacramento, CA, USA, garima.bhandari@csus.edu

D.S.V Bandara is with Department of Mechanical Engineering, Faculty of Engineering, Kyushu University, Fukuoka, Japan, d-sanjaya@mech.kyushu-u.ac.jp

In continuous kinematic prediction, methodologies generally fall into three categories: forecasting from past kinematics, fusing multimodal inputs (e.g., sEMG and IMUs), or directly estimating kinematics from sEMG [8]. The first two approaches suffer from practical limitations: past kinematics lack the physiological "intent" signal needed for sudden directional changes, while multimodal fusion significantly increases computational complexity, memory, and power consumption [9]. Consequently, direct sEMG-to-kinematics mapping remains the preferred approach for deploying streamlined, resource-efficient prosthetic interfaces.

Early sEMG-to-kinematics mapping relied on traditional algorithms such as Support Vector Machines and Linear Regression, which require extensive manual feature engineering and struggle with complex, nonlinear muscle dynamics [10]. Deep learning has since enabled model-free continuous regression by automatically extracting hierarchical features [11], [12]. While LSTM networks became the standard for temporal modeling [13], pure LSTMs struggle to extract robust spatial features directly from raw high-dimensional sEMG. Conversely, Transformer architectures offer powerful self-attention mechanisms [14], [15]; however demand prohibitive memory resources and exhibit quadratic computational complexity, rendering them less viable for low-latency edge deployment [16]. Therefore, a computationally efficient architecture leveraging spatial and temporal features is critically needed.

To address this, we propose a novel Hybrid CNN-LSTM architecture for the continuous regression of 10 simultaneous finger joint kinematics directly from raw sEMG signals. Unlike prior hybrid works targeting wrist or fewer degrees of freedom, this work extends high-fidelity regression to a higher-dimensional joint space while eliminating multimodal sensor dependency. Crucially, a strict repetition-wise evaluation protocol is adopted to prevent temporal data leakage inherent in conventional random-split approaches. Our contributions are *three-fold*: *First*, the proposed model outperforms standard recurrent baselines (LSTM and BiLSTM) in regression accuracy while maintaining computational efficiency. *Second*, we utilize a rigorous repetition-wise training protocol to ensure the temporal generalization of robust muscle patterns over mere memorization. *Third*, we provide a comprehensive performance analysis encompassing varying input windows (window-wise), architectural comparisons (model-wise), and individual digit tracking fidelity (joint-wise).

The remainder of this paper is organized as follows. Section II details the related data and processing methodology. Section III presents the mathematical and architectural details of the proposed Hybrid CNN-LSTM Regressor model, followed by results and discussion in Section IV, and Section V provides the concluding remarks and outlines future research.

## II. Related Data and Processing

We used an existing dataset, named NinaPro DB1 [17], which serves as a benchmark for sparse sEMG-based kinematic estimation in order to ensure the reproducibility and clinical relevance of our study. The dataset contains data from 27 intact subjects. The sEMG signals were acquired using 10 Otto Bock MyoBock 13E200 active electrodes, while the corresponding kinematic data (finger joint angles) were recorded using a 22-sensor CyberGlove II data glove.

### A. Data Acquisition and Protocol

The sEMG electrodes were placed on the forearm. Eight of them were equally spaced around the radio-humoral joint, and two were positioned on the main activity spots of the flexor and extensor digitorum superficialis muscles. The dataset captures 52 movements. However, this study focuses specifically on 10 functional grasp movements essential for daily living (e.g., cylindrical grasp, pinch, tripod grasp). Furthermore, our analysis only considered 10 important finger joint angles associated with these movements.

Each grasp was repeated 10 times by each subject. In order to maintain rigorous evaluation standards, we employed a *repetition-wise split*, allocating 8 repetitions to training, 1 to validation, and 1 to testing. This split ensures that the model is evaluated on entirely unseen gesture instances, rigorously testing its temporal generalization capabilities.

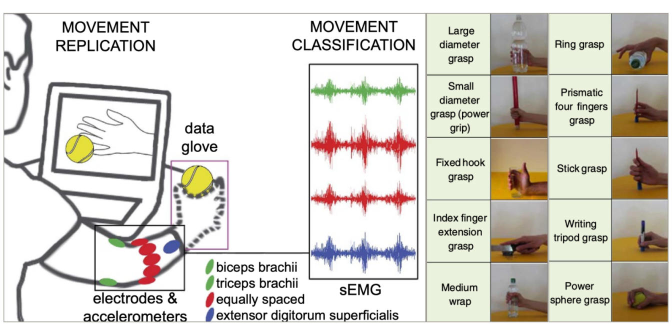


Fig. 1. Data acquisition setup with sEMG sensor and data glove alongside the 10 functional grasp movements considered in this study [17].

### B. Preprocessing, Feature Extraction and Postprocessing

*1) Preprocessing:* Segments corresponding to rest conditions, where participants held a neutral hand posture, were removed from the synchronized sEMG and glove recordings so that the analysis focused only on active muscle contractions and kinematic transitions. The remaining sEMG signals were processed using a sliding-window scheme with window length $T_w$= 800 ms. Each window was partitioned into 20 temporal steps, corresponding to 40 ms of muscle activity per step. To enlarge the training set while maintaining temporal consistency, windows were advanced with a stride of 80 ms (2 time steps), producing a 720 ms (90%) overlap between successive windows. Physiological latency due to Electromechanical Delay (EMD), defined as the delay between electrical muscle activation and observable motion, was compensated through a target shift of $\Delta t = 100\,\mathrm{ms}$. Accordingly, an input sEMG segment covering $[t,\, t + 800\,\mathrm{ms}]$ was used to predict joint angles at $t + 900\,\mathrm{ms}$.

*2) Feature Extraction:* We extracted two computationally efficient, robust-to-noise time-domain features per channel for each time step within a window. They are Mean Absolute Value (MAV) and Root Mean Square (RMS). These features effectively capture the amplitude and energy of muscle contraction, which correlate strongly with the generated force and joint kinematics.

MAV is calculated as the average of the absolute signal amplitude within a window of $N$ samples:

$$\text{MAV} = \frac{1}{N}\sum_{i=1}^{N}|x_i| \tag{1}$$

where $x_i$ represents the $i$-th sample of the sEMG signal.

RMS represents the signal power and is defined as:

$$\text{RMS} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}x_i^2} \tag{2}$$

Given $C$ input sEMG channels, extracting these two features results in a combined feature vector $x_t \in \mathbb{R}^{2C}$ for each time step $t$. Ultimately, the extracted feature sequence serves as the direct input to our proposed Hybrid CNN-LSTM Regressor model, yielding a sequence tensor of shape $(B, L, 2C)$, where B represents the batch size and L represents the sequential time steps.

*3) Postprocessing:* Finally, during the evaluation phase, a Savitzky-Golay filter [18] (window length of 15, which was determined to be the optimal size after empirical evaluation, polynomial order of 3) was applied to the predicted joint angle trajectories of the test set. This post-processing step effectively mitigated high-frequency noise and mechanical jitter. This ensured that the final output kinematics were smooth and physiologically realistic for continuous proportional control.

## III. Proposed Hybrid CNN-LSTM Regressor

In this study, we introduce a Hybrid CNN-LSTM Regressor, a deep learning architecture designed to capture both local feature representations and long-range temporal dependencies of muscle activation for robust kinematic regression.

### A. Model Architecture

The proposed Hybrid CNN-LSTM Regressor integrates 1D Convolutional Neural Networks for spatial feature extraction with Long Short-Term Memory networks for sequential temporal modeling. This architecture is designed to map the

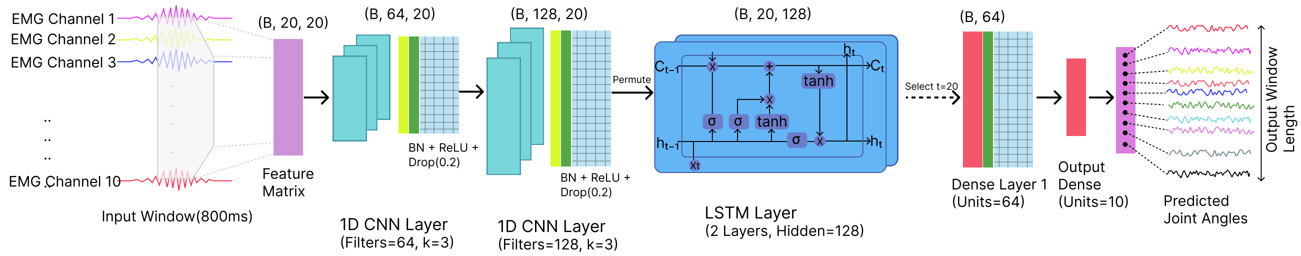


Fig. 2. Architecture of the proposed Hybrid CNN-LSTM.

extracted electromyographic features directly to continuous kinematic trajectories. As illustrated in Fig. 2, the model processes a sequence of $L$ time steps (e.g., $L = 20$ for an 800 ms window). A 2-stage CNN feature extractor first projects the input into a latent space to capture local patterns, followed by a 2-layer LSTM network to model temporal evolution. Finally, the last hidden state ($h_L$) is passed to a fully connected regression head to predict the joint angles.

*1) CNN Feature Extractor:* Let $X_{input} \in \mathbb{R}^{L \times F}$ represent the input feature sequence for a given observation window, where $L$ denotes the number of discrete time steps and $F$ represents the total number of features (MAV and RMS across $C$ sEMG channels). To apply convolutions across the temporal domain, the input is first transposed to the shape $(F, L)$, treating the features as input channels and allowing the convolutional kernels to slide over the time steps.

The sequence is processed by a 2-stage 1D Convolutional block to embed the noisy input features into a latent dimension. Each stage applies a 1D Convolution, followed by Batch Normalization (BN), ReLU activation, and Dropout. Mathematically, the 1D convolution operation extracts spatial-temporal features from the input sequence $X$ by applying a learnable kernel $\mathcal{K}$. For a given position $i$ along the temporal axis, the operation is defined as:

$$\mathcal{S}(i) = (X * \mathcal{K})(i) = \sum_m X(m)\mathcal{K}(i - m) \tag{3}$$

where $\mathcal{S}(i)$ represents the output feature map. Through these sequential transformations, the CNN block effectively smooths local signal noise and extracts higher-level representations. The output is then transposed back, yielding the final embedded sequence $X_{emb}$:

$$X_{emb} = \text{CNNBlock}(X_{input}^T)^T \tag{4}$$

*2) LSTM Temporal Modeling:* To capture the long-term temporal evolution of the muscle signals, the embedded sequence $X_{emb}$ is passed through a stacked LSTM network. LSTMs utilize memory cells and gating mechanisms to selectively retain and discard information over time, effectively overcoming the vanishing gradient issues inherent in traditional recurrent networks.

At each time step $t \in \{1, \ldots, L\}$, the LSTM unit updates its hidden state $h_t$ and cell state $c_t$ based on the current embedded input vector $x_t$ (where $x_t \in X_{emb}$) and the previous hidden state $h_{t-1}$. The governing equations for the forget gate $f_t$, input gate $i_t$, and output gate $o_t$ are formulated as:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{5}$$
$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{6}$$
$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{7}$$

where $W$ and $U$ represent the learnable weight matrices, $b$ denotes the bias vectors, and $\sigma$ is the Sigmoid activation function. The candidate cell state $\tilde{c}_t$ and the final updated cell memory $c_t$ are computed using:

$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{8}$$
$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{9}$$

where $\odot$ denotes the element-wise Hadamard product. Finally, the hidden state output is calculated as:

$$h_t = o_t \odot \tanh(c_t) \tag{10}$$

This continuous aggregation of past and present information allows the model to learn the dynamic temporal trajectory of the hand movement.

*3) Regression Head:* Unlike global average pooling approaches, our architecture utilizes only the *last hidden state* ($h_L$) of the processed sequence to drive the final prediction, as it encapsulates the most recent and cumulative temporal context of the observation window.

$$y_{pred} = \text{FC}(h_L) \tag{11}$$

where FC) denotes the final regression head, which consists of a fully connected multi-layer perceptron with an intermediate ReLU activation and Dropout. The resulting output vector $y_{pred}$ represents the predicted angles for the 10 finger joints at the target time step. All specific layer dimensions and training hyperparameters are detailed in Table I.

## B. Performance Metrics

To evaluate the regression performance, we utilize three standard metrics: Pearson Correlation Coefficient (PCC), Root Mean Square Error (RMSE), and the Coefficient of Determination ($R^2$).

TABLE I
HYPERPARAMETERS AND ARCHITECTURAL TENSOR CONFIGURATIONS OF THE EVALUATED MODELS

| Parameter (Notation) | LSTM | BiLSTM | Hybrid CNN-LSTM |
|---|---|---|---|
| Batch Size ($B$) | 512 | 512 | 512 |
| Training Epochs | 100 | 100 | 100 |
| Optimizer | Adam | Adam | Adam |
| Learning Rate | 0.001 | 0.001 | 0.001 |
| Sequence Time Steps ($L$) | 10, 15, 20 | 10, 15, 20 | 10, 15, 20 |
| Input Features per Step ($F$) | 20 | 20 | 20 |
| CNN Feature Extractor | None | None | 2× Conv1D (64, 128), $\mathcal{K} = 3$ |
| CNN Output Shape | None | None | $(B, 128, L)$ |
| Hidden State Size ($h_{dim}$) | 128 | 128 (per dir.) | 128 |
| LSTM Output Shape ($h_L$) | $(B, 128)$ | $(B, 256)$ | $(B, 128)$ |
| Dense Layer (FC) | 128→64→10 | 256→64→10 | 128→64→10 |
| Dropout Rate ($p$) | 0.2 | 0.2 | 0.2 |
| Final Output Shape ($y_{pred}$) | $(B, 10)$ | $(B, 10)$ | $(B, 10)$ |

*1) Pearson Correlation Coefficient (PCC):* PCC measures the strength of the linear relationship between the predicted ($\hat{y}$) and ground truth ($y$) trajectories:

$$\text{PCC} = \frac{\sum(\hat{y}_i - \bar{\hat{y}})(y_i - \bar{y})}{\sqrt{\sum(\hat{y}_i - \bar{\hat{y}})^2 \sum(y_i - \bar{y})^2}} \quad (12)$$

*2) Root Mean Square Error (RMSE):* RMSE quantifies the average magnitude of the prediction error in degrees:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \quad (13)$$

*3) Coefficient of Determination ($R^2$):* $R^2$ represents the proportion of variance in the dependent variable that is predictable from the independent variable:

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2} \quad (14)$$

## IV. RESULTS AND DISCUSSION

In this section, we evaluate the performance of the proposed Hybrid CNN-LSTM Regressor against standard baseline architectures (Standard LSTM and Bidirectional LSTM).

### A. Performance for Varying Input Window Sizes

In this section, the performance of the proposed Hybrid CNN-LSTM Regressor model and the baseline models (LSTM and BiLSTM) is investigated for three input $T_w$ (400, 600, and 800 ms). A comprehensive summary of all evaluation metrics ($R^2$, RMSE, and PCC) across the different architectures and temporal windows is detailed in Table II.

*1) Coefficient of Determination ($R^2$):* : The comparison of the average $R^2$ along with standard deviation for LSTM, BiLSTM, and Hybrid CNN-LSTM Regressor models across the three sliding windows is shown in Fig. 3. To evaluate the statistical significance of the performance differences between the architectures, paired t-tests were conducted.

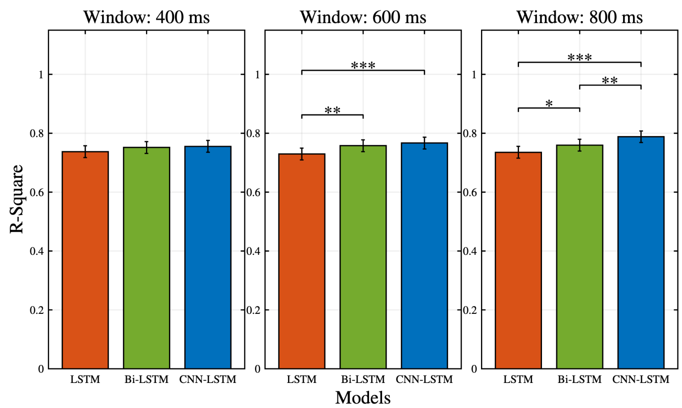


Fig. 3. Average $R^2$ Score over the testing dataset for input $T_w$ of 400 ms, 600 ms, and 800 ms. Statistical significance from paired t-tests was evaluated at a significance level of $\alpha = 0.05$ and is denoted as: * ($p < 0.05$), ** ($p < 0.01$), and *** ($p < 0.001$). Comparisons yielding non-significant differences ($p > 0.05$) remain unmarked.

For $T_w$=400 ms, the standard LSTM shows an average $R^2$ of $0.7347 \pm 0.0290$, while the BiLSTM performs better than it with $0.7451 \pm 0.0252$. The proposed Hybrid CNN-LSTM regressor model achieves the highest score of $0.7570 \pm 0.0250$. The performance differences between the models are not statistically significant at this short $T_w$.

On increasing the $T_w$ to 600 ms, the LSTM performance dips slightly ($0.7337 \pm 0.0286$), which suggests difficulties in capturing longer temporal dependencies without spatial feature extraction. However on the contrary, the Hybrid CNN-LSTM regressor demonstrates improved consistency, rising to $0.7710 \pm 0.0191$. At 600 ms, both the BiLSTM ($p < 0.01$) and the Hybrid CNN-LSTM regressor ($p < 0.001$) show statistically significant improvements over the standard LSTM. However, the difference between the hybrid model and BiLSTM is not yet significant.

Finally, at $T_w$=800 ms, the Hybrid CNN-LSTM regressor achieves its peak performance with an $R^2$ of $0.7733 \pm 0.0207$. At this temporal depth, the Hybrid CNN-LSTM regressor

TABLE II
COMPREHENSIVE PERFORMANCE COMPARISON OF DEEP LEARNING ARCHITECTURES (TEST SET – REPETITION 10)

| Model Architecture | Window Size ($T_w$) | Time Steps ($L$) | $\mathbf{R^2}$ | RMSE (°) | PCC |
|---|---|---|---|---|---|
| LSTM | 400 ms | 10 | $0.7347 \pm 0.0290$ | $12.5497 \pm 2.4385$ | 0.8602 |
| | 600 ms | 15 | $0.7337 \pm 0.0286$ | $12.6283 \pm 2.4385$ | 0.8543 |
| | 800 ms | 20 | $0.7408 \pm 0.0227$ | $12.4618 \pm 2.4355$ | 0.8577 |
| BiLSTM | 400 ms | 10 | $0.7451 \pm 0.0252$ | $12.3077 \pm 2.3940$ | 0.8689 |
| | 600 ms | 15 | $0.7518 \pm 0.0278$ | $12.1852 \pm 2.3198$ | 0.8708 |
| | 800 ms | 20 | $0.7518 \pm 0.0290$ | $12.1719 \pm 2.3071$ | 0.8721 |
| Hybrid CNN-LSTM | 400 ms | 10 | $0.7570 \pm 0.0250$ | $12.0083 \pm 2.2803$ | 0.8698 |
| | 600 ms | 15 | $0.7710 \pm 0.0191$ | $11.7043 \pm 2.1637$ | 0.8759 |
| | 800 ms | 20 | $0.7733 \pm 0.0207$ | $11.6454 \pm 2.2225$ | 0.8841 |

significantly outperforms both the standard LSTM ($p < 0.001$) and the BiLSTM ($p < 0.01$). Additionally, the BiLSTM maintains a significant advantage over the standard LSTM ($p < 0.05$). This trend confirms that longer temporal windows allow the hybrid model to capture the underlying muscle activation dynamics more accurately, leading to more precise kinematic mapping.

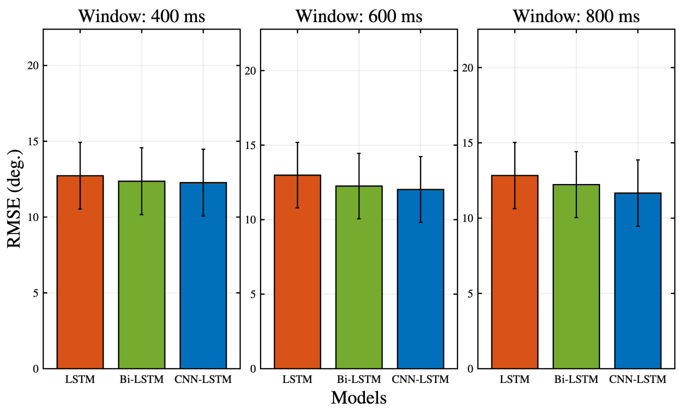


Fig. 4. Average RMSE (°) over the testing dataset across architectures for input $T_w$ of 400 ms, 600 ms, and 800 ms. The absence of significance markers indicates that the differences between the models are not statistically significant ($p > 0.05$ for all comparisons).

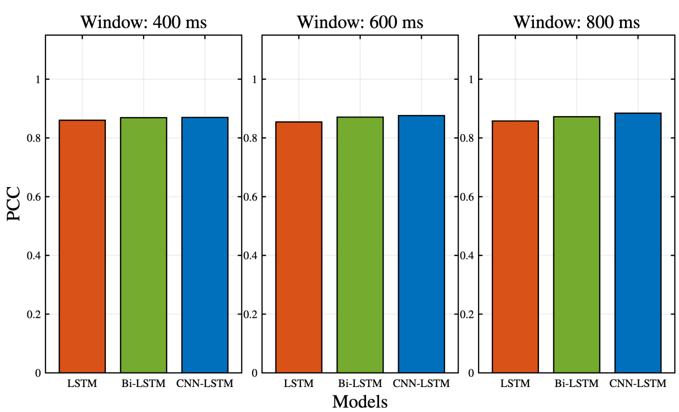


Fig. 5. Average Pearson Correlation Coefficient (PCC) over the testing dataset across architectures for input $T_w$ of 400 ms, 600 ms, and 800 ms.

*2) Root Mean Square Error (RMSE):* The RMSE comparison is shown in Fig. 4. Similar to the $R^2$ trend, the Hybrid CNN-LSTM regressor consistently achieves the lowest mean error rates. At $T_w$=400 ms, the standard LSTM records an RMSE of $12.55° \pm 2.44°$. At the same time, for the proposed Hybrid CNN-LSTM regressor, the error reduces to $12.01° \pm 2.28°$, indicating improved robustness in capturing local muscle activation patterns. On increasing the $T_w$ to 600 ms, the performance gap becomes more evident. The LSTM yields an RMSE of $12.63° \pm 2.44°$, whereas the Hybrid CNN-LSTM regressor achieves a lower error of $11.70° \pm 2.16°$, which demonstrates the advantage of combining convolutional feature extraction with temporal modeling. At the longest temporal window ($T_w = 800$ ms), the Hybrid CNN-LSTM regressor achieves its best performance with an RMSE of $11.65° \pm 2.22°$, outperforming both baseline recurrent models.

It should be noted that despite the observable reductions in mean error, statistical analysis shows that the differences in RMSE between the models for all three $T_w$ are not statistically significant ($p > 0.05$). This is primarily due to the relatively high standard deviations, which reflect the natural variance in error bounds across the 10 different finger joints (e.g., highly active index fingers versus coupled, passive little fingers). Regardless, the consistent reduction in the average error emphasizes the Hybrid model's robustness against the noise inherent in sEMG signals.

*3) Pearson Correlation Coefficient (PCC):* Fig. 5 illustrates the average PCC for the evaluated architectures across the three input window sizes. The PCC metric evaluates the linear correlation between the predicted and actual joint angle trajectories, reflecting how well the model captures the overall shape and timing of the user's intended movements.

At $T_w$=400 ms, the standard LSTM yields a PCC of 0.8602. The BiLSTM and Hybrid CNN-LSTM regressor perform comparably at this stage, achieving scores of 0.8689 and 0.8698, respectively. However, as the temporal window size increases, the advantage of the hybrid architecture becomes more evident. At 600 ms, the Hybrid CNN-LSTM regressor improves to 0.8759, whereas the LSTM slightly drops to 0.8543. The proposed hybrid model achieves its maximum correlation of 0.8841 at the 800 ms window, noticeably outperforming both the standard LSTM (0.8577) and the BiLSTM (0.8721). This confirms that combining CNN-based spatial filtering with LSTM sequence modeling enables the system to highly synchronize with the natural, dynamic joint trajectories.

### *B. Joint-Wise Trajectory Comparison*

To provide a more intuitive, fine-grained evaluation of the proposed model, we analyzed the continuous regression trajectories for each of the 10 finger joints independently. The actual joint angle curves(Ground Truth) alongside the predicted trajectories generated by the standard LSTM, BiLSTM, and the proposed Hybrid CNN-LSTM regressor models for a representative test sequence (Subject 13, Repetition 10) are shown in Fig. 6.

While the standard LSTM and BiLSTM models successfully capture the macroscopic motion trends, their prediction curves fluctuate more significantly during rapid grasp transitions. These pure recurrent models exhibit pronounced overshoots, underestimations, or jitter at peak flexion and extension angles frequently. This instability is particularly evident in highly active joints with a wide range of motion, such as the Thumb Flexion, Index Distal, and Ring Distal joints. In contrast, the proposed Hybrid CNN-LSTM model maintains high-fidelity tracking with minimal latency, closely following the rapid directional changes of the ground truth. Furthermore, for passive or coupled joints (such as the Little Proximal and Distal joints), which inherently exhibit higher noise levels due to the mechanical coupling of the data glove sensors, the CNN-LSTM successfully filters out local motion artifacts.

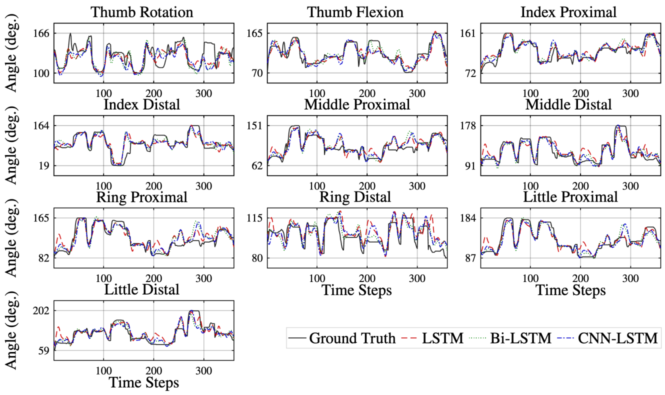


Fig. 6. Actual vs. predicted trajectories for Subject 13(Test Set) across 10 finger joints, comparing the LSTM, BiLSTM, and proposed CNN-LSTM.

It completely avoids the unstable oscillations common to standard RNN approaches. Therefore, it yields a substantially smoother, realistic kinematic trajectory.

Comparatively, the proposed model demonstrates highly competitive performance across all degrees of freedom. While landmark proportional control studies achieved average correlations (PCC) of 0.79 for finger kinematics [7], our hybrid model peaks at a PCC of 0.88. Moreover, compared to recent hybrid architectures for wrist kinematics [19] or continuous state estimators [6], we extend high-fidelity regression (RMSE $\approx 11.65°$) to the much higher-dimensional problem of tracking 10 independent finger joints. Although direct cross-study comparisons require caution due to differing experimental protocols, sensor types, and target variables (e.g., velocities versus absolute angles), these results confirm that integrating spatial and temporal deep learning provides a robust framework for continuous intention decoding.

## V. Conclusion

In this study, we developed a Hybrid CNN-LSTM Regressor model and evaluated it for continuous regression of 10 finger joint kinematics directly from raw sEMG signals. A rigorous, repetition-wise data-splitting protocol was implemented using the open-access Ninapro DB1 dataset to ensure robust evaluation. This refined the kinematic mapping process without relying on complex, computationally heavy multimodal sensor fusion. The experimental results demonstrate that the proposed Hybrid CNN-LSTM Regressor model consistently outperforms standard baseline models (LSTM and BiLSTM). A comprehensive analysis of variable input $T_w$, established that an 800 ms temporal window provides the optimal context for muscle activation dynamics, achieving the highest overall accuracy ($R^2 = 0.7733$, PCC $= 0.8841$) and the lowest error profile (RMSE $= 11.65°$). This spatial-temporal integration enables robust control using solely sEMG inputs, showing significant promise for deployment in low-latency, resource-constrained prosthetic controllers. Future work will integrate attention mechanisms, enhance computational efficiency, apply transfer learning to mitigate inter-subject variability, and explore window sizes beyond 800 ms, ultimately maximizing clinical viability across diverse users.

## References

[1] M. Schouten, E. C. Baars, U. S. Yavuz, and G. Krijnen, "Joint angle and torque estimation using multifrequency electrical impedance myography and surface electromyography," *IEEE Sensors Journal*, vol. 24, no. 20, pp. 32 651–32 659, 2024.

[2] Z. Tang, X. Deng, Y. Wen, X. Han, J. Wu, and Z. Yu, "Cross-activity semg-driven joint angle estimation via hybrid attention fusion: Bridging traditional features and deep spatial representations," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2025, pp. 14 611–14 616.

[3] R. Mio, J. Narayan, and A. A. Faisal, "Motor unit decomposition over intrinsic hand muscles during single and multi-finger flexion," in *Summer School on Neurorehabilitation*. Springer, 2022, pp. 115–119.

[4] S. Pratap, J. Narayan, Y. Hatta, K. Ito, and S. M. Hazarika, "Glove-net: Enhancing grasp classification with multisensory data and deep learning approach," *Sensors*, vol. 24, no. 13, 2024.

[5] S. Pratap, A. Gupta, J. Narayan, K. Ito, and S. M. Hazarika, "Data-driven grasp posture recognition with data gloves and machine learning models," in *Design and Control of Rehabilitation Robots: From Concept to Therapy*. Springer, 2025, pp. 231–246.

[6] H. Zhang, B. Peng, L. Tian, O. W. Samuel, and G. Li, "Continuous kalman estimation method for finger kinematics tracking from surface electromyography," *Cyborg and Bionic Systems*, vol. 5, p. 0094, 2024.

[7] M. Xiloyannis, C. Gavriel, A. A. C. Thomik, and A. A. Faisal, "Gaussian process autoregression for simultaneous proportional multi-modal prosthetic control with natural hand kinematics," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 10, pp. 1785–1801, 2017.

[8] C. Lin and S. Zhou, "A parallel network for continuous motion estimation of finger joint angles with surface electromyographic signals," *Applied Sciences*, 2024.

[9] O. Karaali, H. Farag, S. Dosen, and C. Stefanovic, "Enabling next-generation cloud-connected bionic limbs through 5g connectivity," *IEEE Internet of Things Magazine*, vol. 8, no. 4, pp. 45–51, 2025.

[10] L. Ling, Z. Lin, B. Feng, L. Wei, L. Jin, and Y. Wang, "Continuous prediction of human knee joint angle using a sparrow search algorithm optimized random forest model based on semg signals," *Automatika*, vol. 66, no. 2, pp. 321–334, 2025.

[11] C. Lin, X. Zhang, and C. Zhao, "A parallel and efficient transformer deep learning network for continuous estimation of hand kinematics from electromyographic signals," *Scientific Reports*, vol. 15, no. 1, p. 36150, 2025.

[12] J. Narayan, A. Mishra, and H. Gritli, "Pathogaitnet: A deep temporal model for predicting pathological gait trajectories in pediatric patients," in *2025 11th International Conference on Control, Decision and Information Technologies (CoDIT)*. IEEE, 2025.

[13] C. Wang, W. Guo, C. Lin *et al.*, "semg-based continuous estimation of grasp movements by long-short term memory network," *Biomedical Signal Processing and Control*, vol. 60, p. 101970, 2020.

[14] Z. Chen, H. Wang, H. Chen, and T. Wei, "Continuous motion finger joint angle estimation utilizing hybrid semg-fmg modality driven transformer-based deep learning model," *Biomedical Signal Processing and Control*, vol. 85, p. 105030, 2023.

[15] N. A. S. Putro, C. Avian, S. W. Prakosa, M. I. Mahali, and J.-S. Leu, "Estimating finger joint angles by surface emg signal using feature extraction and transformer-based deep learning model," *Biomedical Signal Processing and Control*, vol. 87, p. 105447, 2024.

[16] N. Mehlman, J.-C. Gagnon-Audet, M. Shvartsman, K. Niu, A. H. Miller, and S. Sodhani, "Scaling and distilling transformer models for semg," *arXiv preprint arXiv:2507.22094*, 2025.

[17] M. Atzori, A. Gijsberts, C. Castellini, B. Caputo, A.-G. M. Hager, S. Elsig, G. Giatsidis, F. Bassetto, and H. Muller, "Electromyography data for non-invasive naturally-controlled robotic hand prostheses," *Scientific data*, vol. 1, no. 1, p. 140053, 2014.

[18] A. Savitzky and M. J. Golay, "Smoothing and differentiation of data by simplified least squares procedures." *Analytical chemistry*, vol. 36, no. 8, pp. 1627–1639, 1964.

[19] J. Du, Z. Liu, W. Dong, W. Zhang, and Z. Miao, "A novel tcn-lstm hybrid model for semg-based continuous estimation of wrist joint angles," *Sensors*, vol. 24, no. 17, p. 5631, 2024.