SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V11I11P106

Volume 11 Issue 11, 53-64, November 2024

Original Article

Attention-Driven Hybrid LSTM-GRU Model for Enhanced EMG Signal Hand Gesture Recognition

Ranjeesh R. Chandran[1], D. Devaraj[2], Sreedeep Krishnan[3]

[1,3]*Department of Instrumentation and Control Engineering, Kalasalingam Academy of Research and Education, Tamilnadu, India.*
[2]*School of Electronics and Electrical Technology, Kalasalingam Academy of Research and Education,Tamilnadu, India.*

[1]*Corresponding Author : ranjeesh.r@gmail.com*

Received: 04 September 2024 Revised: 05 October 2024 Accepted: 03 November 2024 Published: 30 November 2024

Abstract - *Hand Gesture Recognition (HGR) is crucial for Human Computer Interaction (HCI) with applications including assistive technologies for disabled persons to advanced human computer interfaces. HGR from Electromyography (EMG) signals possessed issues regarding noise and variability due to the complex muscle movements. So, for effectively recognizing hand gestures from EMG signals, this paper proposes a hybrid Deep Learning (DL) model combining Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks, incorporating an attention mechanism to highlight pertinent features and enhance the model's focus on critical data segments. The study utilized a dataset with static hand movements captured using an MYO Thalmic bracelet with eight equally distributed sensors, and the raw EMG data was preprocessed. Feature extraction is analyzed in both the time and frequency domains, leading to a more robust and comprehensive analysis of the EMG signal. The proposed model achieved 98.875% accuracy, 97.82% precision, 98.07% recall, and 97.74% F1 score, outperforming existing models. Thus, the proposed model demonstrates significant advancements in hand gesture recognition with high accuracy, making it reliable for several real-time applications.*

Keywords - *Hand Gesture Recognition, Human Computer Interaction, Electromyography, MYO Thalmic, Long Short-Term Memory, Attention mechanism gated, Recurrent unit.*

1. Introduction

A hand gesture is a significant and explanatory posture or actual movement of the hands. Hand gestures are considered one of the important ways for humans to convey and express intuitive intentions like emotions, thoughts and ideas without words [1]. Hand gestures are used in deaf and speech-impaired people communication, Human-Computer Interaction (HCI), robot control, home automation, clinical applications, sign language recognition, problem-solving, decision-making, and understanding of complex subjects. Different types of gestures include finger movements and hand grasps. Different finger movements are possible due to the flexion produced by each finger [2]. The human body consists of two large muscles responsible for each finger's flexion and additional muscles supporting their movements. The basic hand/finger gestures are given in Figure 1.

The HGR is broadly divided into static and dynamic recognition. Static HGR is based on a single image or frame, like a thumbs-up or peace sign, where the gesture is identified by its particular shape or configuration [3]. On the other hand, dynamic methods use the temporal features of hand motion by considering the gesture as a sequence of hand shapes. According to various applications, static and dynamic HGRs are critical as static HGRs are employed in command-based systems, and dynamic HGRs offer more expressive and organic interactions [4].

Fig. 1 Basic hand gesture

Unlike gesture detection, which primarily identifies the presence of gestures, HGR focuses on interpreting the meaning. Motion tracking involves following the trajectory of movements but may not infer the intent behind them. Conventional HGR commonly utilizes cameras, data gloves and Inertial Measurement Units (IMU). Data gloves and IMU necessitate direct contact with the fingers to track hand joints. However, limiting natural interaction and cameras require a direct line of sight. These technologies excel at recording motions but fail to capture subtle kinetic changes in hand movements, especially during muscle fatigue [5, 6].

Electromyography (EMG) based HGR is an emerging field in HCI and biomedical engineering. The EMG signal is a random signal that relies on the muscles' neuromuscular, anatomical and physiological properties [7]. These signals are captured by placing surface electrodes on the skin, providing important information about the muscle movements. Surface EMG (sEMG) is a non-invasive method of acquisition of EMG signal using surface electrodes. The amplitudes of the sEMG hang on the muscle under study and the characteristics of electrodes. Recent advancements in EMG-based HGR enhance the accuracy and reliability of gesture classification [8]. The applications included assistive technologies for persons with physical disabilities, which can be used as substitutes for prosthetics and other digital systems. In addition, as the EMG signal propagates through the tissues, noise gets added to the signal, which can be recorded using different types of electrodes placed on the skin. The recorded EMG signals are processed for further analysis.

Given wide-ranging applications, there is a need for developing cutting-edge recognition methods for accurate and versatile recognition of gestures, paving the way for technological advancements in areas such as virtual and augmented reality, touchless control interfaces, sign language recognition, leading to improved quality of life and enhancements in human performances. Thus, this paper uses a hybrid DL method to generate HGR from EMG signals efficiently. The main contributions of the study are as follows.

- To improve recognition accuracy in hand gesture recognition by effectively processing EMG signals.
- Analyzing signals in both time and frequency domains to improve gesture recognition systems' performance.
- To develop a hybrid DL model utilizing LSTM and GRU networks to capture EMG signals' temporal dependencies.
- Integrate the attention mechanism to emphasize pertinent features of the EMG signal for accurate gesture recognition.

The structure of the study unfolds as follows: Section 2 provides related works of hand gesture recognition methods and explores the research gap that delves into the proposed model. Section 3 details the proposed hybrid deep learning model enhanced with the attention mechanism. Section 4 details the in-depth analysis of the results of handling EMG signals. Section 5 furnishes concluding remarks.

2. Literature Review

Lopez et al. [9] employed spectrograms and Convolutional Neural Networks (CNN) to analyze the effects of post-processing, which filtered prediction sequences and removed false labels on the HGR model. The effect of memory cells on model accuracy was evaluated by comparing the performance of CNN and CNN-LSTM. The EMG-EPN -612 dataset with five hand motions was utilized. With post-preprocessing, the CNN-LSTM model achieved an accuracy of 90.55%. The model possessed limitations due to the computational load from LSTM networks. Pourmokhtari et al. [10] investigated sEMG signals for controlling upper arm prostheses utilizing four unique channels to classify five different finger movements. The study employed k-nearest neighbors for the classification and two-time domain features (min, max) with Root Mean Square (RMS) and Mean Absolute Value (MAV). The combination of min and max with MAV yielded the highest accuracy rates of 91.0%, 89.9%, 89.8% and 96.0%, respectively, for the four channels. The study possessed challenges in muscle coverage and potential cross talk.

Wang et al. [11] suggested a model to virtually increase the dimension of EMG signals without the need for additional EMG signal acquisition electrodes for accurate gesture recognition by providing the solution for the problem of peak accuracy triggered by the information saturation from physical recordings. The study introduced the Separability of Feature Vectors (SFV), a feature selection method derived from the divergence and correlation of the feature extracted. The effectiveness of increasing the virtual dimension strategy was verified using SFV by predicting the classification result before the classification process. SFV outperformed statistical motion intention recognition models and was suitable for small sample sets.

Abdelaziz et al. [12] established a hybrid CNN-LSTM model to extract features from EMG signals and to capture important features from gestures. The study utilized two publicly available datasets, DualMyo and EMG36, with 8 hand gestures. The testing time issues are solved without compromising the testing accuracy by cascading CNN and max pooling layer to obtain a reduction rate of 1/20 in testing time compared with hybrid CNN-LSTM. The experimental outcomes showed that the suggested method was suitable for real-time applications with challenges, including expanding gesture vocabulary. Zhang et al. [13] suggested a dynamic time regularization approach due to the instability and sensitivity of sEMG signals for accurate and real-time gesture recognition. The similarity between the sample and model was determined by fusing three sEMG signals with the Dynamic Time Warping (DTW) algorithm with a 3D printed prosthesis. Six gesture classification models were combined to generate the optimal feature model. The model achieved an accuracy of 93.75% with challenges including the generalization for prosthetic arms.

Ashraf et al. [14] addressed the limitations in real-time applications because of the latency introduced by the optimization technique. The authors proposed a CNN model based on Myoelectric Control (MEC), utilizing a generalized hyperparameter setting and an improved data segmentation method as a solution. Two different segmentation approaches, disjoint and overlap, were assessed using different overlap sizes. Using a segment size of 200 ms and 80% overlap, the overlap technique significantly outperformed the disjoint method (p-value < 0.05). Hyperparameter tuning uses Bayesian optimization, which reduces the mean Classification Error Rate (CER), outperforming traditional manual, grid, and random search methods.

Al-Khazzar et al. [15] focused on three neural network models, a four layer Deep Neural Network (DNN), an eight-layer DNN and a 5-layer CNN, to identify and classify seven gestures from a public EMG dataset. The study utilized five optimizers, Adam, Adamax, Nadam, Adagrad, and AdaDelta, for testing. Adding layers resulted in slight accuracy improvement, indicating that four-layer DNN offered good performance. The Adam optimizer provided the best results over all models. The model's limitations included the higher computational requirements with increasing layers for models.

Côté-Allard et al. [16] proposed a gesture recognition system that learns informative properties from massive amounts of data generated by combining signals from many users, reducing recoding effort while increasing recognition of gestures. As a result, the study recommended using transfer learning on aggregated data while exploiting the ability of DL algorithms to discover discriminant characteristics from big datasets.

Jain et al. [17] suggested an EMG signal classification model based on ML approaches focused on Support Vector Machine (SVM) and data selection based Genetic Algorithm (GA). SVM was trained using the average similarity values determined using cosine similarity. Rubio et al. [18] used supervised machine learning approaches to create an automatic recognition system for hand or wrist movements. The dataset consists of recordings of EMG signals acquired from 36 subjects. The performance of the Random Forest (RF) model and CNN has been assessed, and the RF model outperformed.

Although hand gesture recognition with EMG signals has made great strides, existing approaches have several drawbacks that require more research and development. Even though hybrid models that combine CNN and LSTM have a good accuracy rate, they frequently have a large computational burden. Techniques based on basic time-domain characteristics and classification algorithms like k-nearest neighbors provide good accuracy but suffer from cross-talk and muscle coverage issues. Techniques that effectively expand the dimensions of the EMG signal complicate selecting and extracting features. Some models exhibit great accuracy and improvements in testing time, but they struggle to increase data size with different gestures. In addition, the sensitivity of sEMG signals remains a significant challenge, emphasizing the need for robust feature extraction and advanced deep learning models to enable real-time applications with low processing costs and enhanced accuracy.

3. Materials and Methods

Hand gesture recognition using EMG signals is crucial, but it also has several challenges, including noise, variability in signal patterns, and the need for real-time processing. The recognition process is difficult because of the complexity of muscle movements and the minute variations in EMG signal patterns. So, a hybrid DL model is suggested for efficient HGR. The EMG signal from the dataset is preprocessed, and features are extracted before training the hybrid attention-based LSTM-GRU model. The gestures are recognized and classified through softmax and fully connected layers. Figure 2 provides the block diagram of the suggested model.

3.1. Dataset

This study utilized data from [19], recording the patterns using a MYO Thalmic bracelet worn on a user's forearm and a PC equipped with a Bluetooth receiver. Equally distributed around the forearm, eight sensors are embedded in the bracelet to acquire the myographic signals. These signals are transmitted to the PC through the Bluetooth interface. The dataset consists of raw EMG data from 36 subjects while performing a series of static hand gestures. The subjects execute two series, with six (seven) fundamental gestures each. Each gesture was achieved for 3 seconds with a gap of 3 seconds between gestures. There are about 40000-50000 recordings in each column. The description of raw data is given in Table 1.

Table 1. Description of class labels for gestures

Class Label	Gesture Description
0	Unmarked Data
1	Hand at Rest
2	The Hand Closed in a Fist
3	Wrist Flexion
4	Wrist Extension
5	Radial Deviations
6	Ulnar Deviations
7	Extended Palm

3.2. Data Preprocessing and Exploratory Data Analysis

Preprocessing of EMG data is crucial to ensuring high-quality input for the hand gesture recognition models. Data normalization is organizing data in a database to remove redundancy and improve data integrity [20]. To guarantee the streamlining and consistency of the stored data, the process involves structuring tables and recognizing relationships between them, which implies simplified data management across all records and fields. Data normalization standardizes the range of EMG signal data using Min- Max normalization, boosting uniformity and aiding convergence during training. Min-Max normalizes and transforms each feature to a range typically between 0 and 1. The process is an alternative to zero mean, unit variance scaling. Mathematically, Equation (1) represents Min-Max scaling,

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{1}$$

Where x is the original value of a feature, x_{min} is the minimum, and x_{max} is the maximum value of the feature in the dataset.

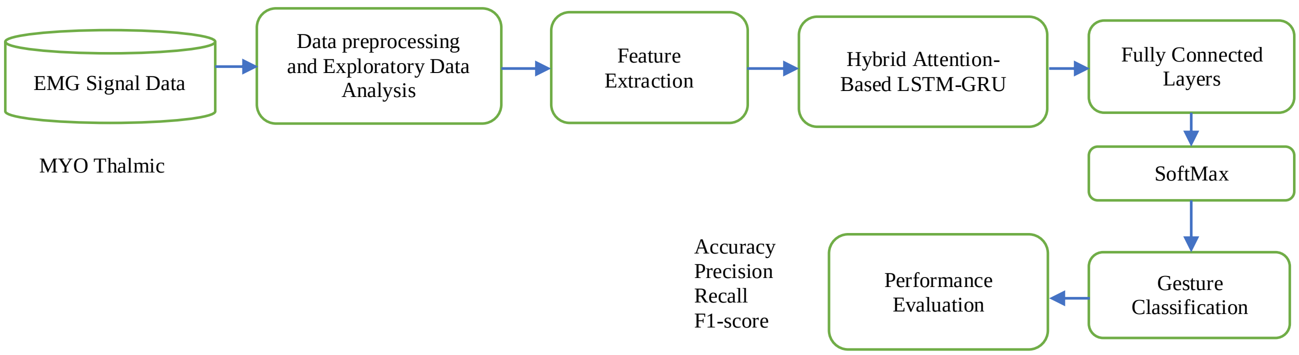

Fig. 2 Proposed model block diagram

By maintaining the relative relationship between the other values, the Min-Max scaler scaled linearly by preserving the effect of outliers, where the maximum value is represented by the largest data point that occurs, and the minimum value is represented by the smallest one. Data cleaning is also done for finding and fixing errors and inconsistencies in data. This stage is important because inconsistent data might lead to false or erroneous conclusions. Data cleaning involves handling missing values, checking for duplicates, and checking for each column's data types and unique values. Exploratory Data Analysis (EDA) is used to analyze the distribution and characteristics of the data. Precise gesture recognition depends on identifying the outliners, anomalies and other data quality issues like missing or duplicate values, which EDA enables. The histogram in Figure 3 illustrates the distribution of data for each channel and the class label in the dataset with EMG signals.

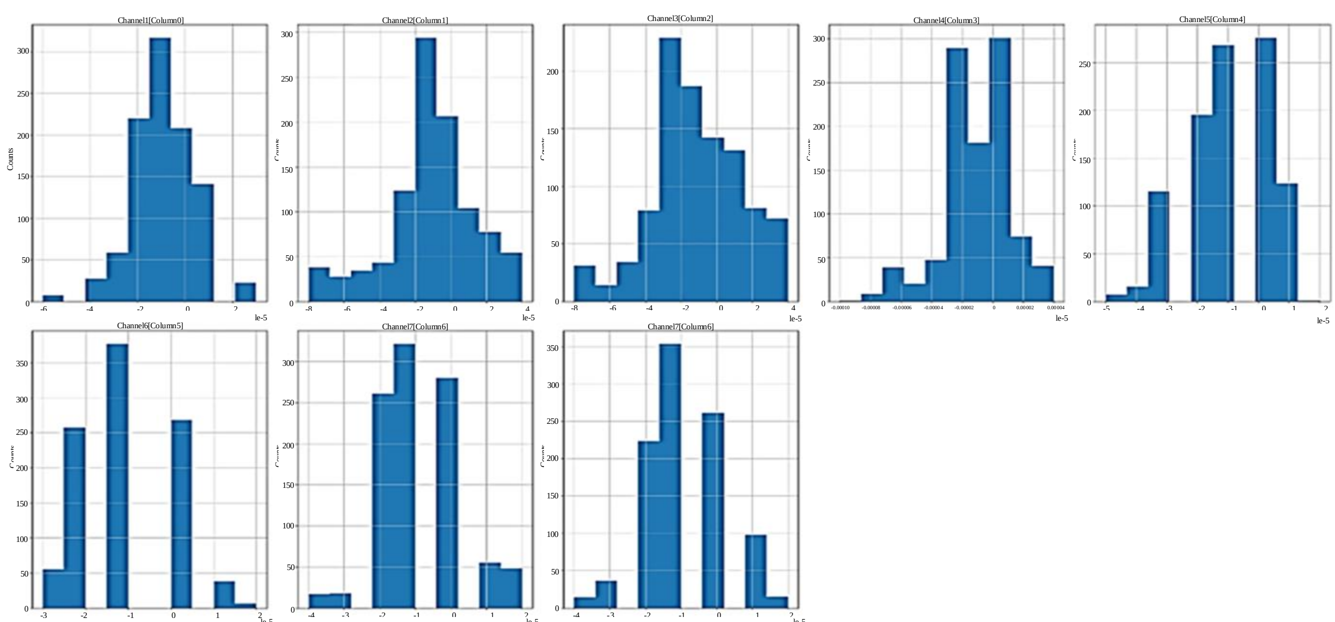

Fig. 3 Histogram of sampled columns

The frequency distribution of the signals obtained through the MYO Thalmic bracelet for channels 1 to 8 is provided. With values ranging from 0 to 7, the histogram for the class label shows the distribution of different gesture classes, as shown in Table 1. Figure 4 illustrates the correlation matrix of the EMG data, providing the visual representation of pairwise correlation coefficients between various channels and time in the dataset. These coefficients show the intensity and direction of the correlations between the variables, which range from -1 to 1. The value 1 denotes a positive correlation, whereas the negative value shows a negative correlation.

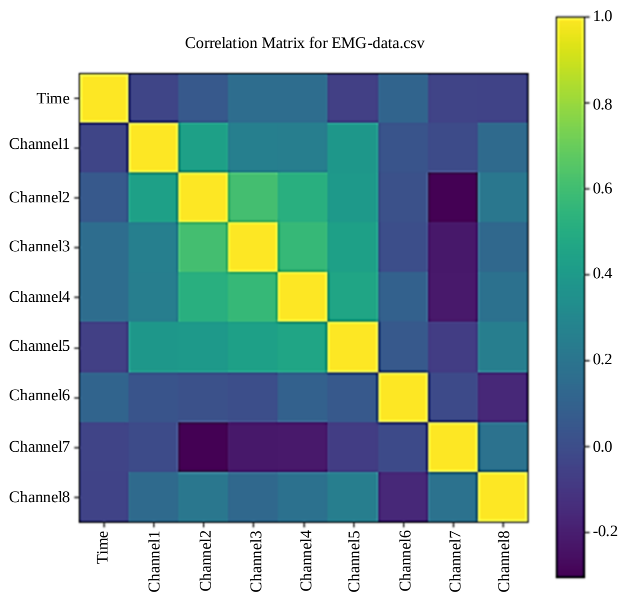

Fig. 4 Correlation matrix of the EMG data

Figure 5 represents the scatter and density plot of the signal, providing relationships and distribution of numerical features in the dataset. The scatter plot displays the pairwise correlation between the different channels. The density plot is given diagonally in the scatter matrix, providing a distribution of individuals where each peak indicates the most common values for each feature.

The time domain signal is transformed into the frequency domain signal using the Fast Fourier Transform (FFT) for frequency domain analysis. The FFT of a discrete signal y(n) is defined by (2).

$$Y(k) = \sum_{n=0}^{N-1} y(n) e^{\frac{-j2\pi kn}{N}} \tag{2}$$

Where N is the number of points in the signal, k is the frequency index, and j denotes the imaginary unit. Filtering using a Low Pass Filter (LPF) is applied after obtaining FFT to remove the unwanted frequencies below a certain cutoff frequency f_c. The low pass filter in the frequency domain is defined by (3).

$$H(k) = \begin{cases} 1 & if\ |f_k| \leq f_c \\ 0 & if\ |f_k| > f_c \end{cases} \tag{3}$$

The filtered signal output, as in Figure 6, is obtained by multiplying the FFT of the signal with filter $H(k)$ as (4).

$$Y_f(k) = Y(k) \cdot H(k) \tag{4}$$

The inverse FFT returns the filtered signal in the frequency domain to the time domain by (5).

$$y_f(n) = \frac{1}{N} \sum_{k=0}^{N-1} Y_f(k) e^{\frac{-j2\pi kn}{N}} \tag{5}$$

The envelope of the original signal is extracted using the Hilbert transform by obtaining the analytic signal as in Figure 7. The analytic signal of the original signal is given by (6).

$$z(t) = y(t) + j\hat{y}(t) \tag{6}$$

Where $\hat{x}(t)$ is the Hilbert transform represented by (7).

$$\hat{y}(t) = \mathcal{H}[y(t)] = \frac{1}{\pi} P \cdot V \int_{-\infty}^{\infty} \frac{x(\tau)}{t-\tau} d\tau \tag{7}$$

Here, $P \cdot V$ is the Cauchy principal value. A signal's envelope is the analytic signal's magnitude, as represented by (8).

$$E(t) = |z(t)| = \sqrt{y(t)^2 + \widehat{y(t)}^2} \tag{8}$$

An LPF is again used to soften the envelope. The low pass filter uses convolution with a smoothing kernel with window size M, as illustrated by (9). The smoothened output of the envelope is shown in Figure 8.

$$E_{smooth}(t) = \frac{1}{2M+1} \sum_{k=-M}^{M} E(t+k) \tag{9}$$

3.3. Feature Extraction

The valuable signal characteristics in the time domain are extracted using Mean Absolute Value (MAV), Root Mean Square (RMS), Waveform Length (WL), Zero Crossing Rate (ZCR), and Simple Square Integral (SSI). These complementary features provided a comprehensive representation of the data, ensuring the model could effectively learn patterns for gesture classification. The effectiveness of this approach lies in its ability to balance the strengths of time and frequency domain analyses. The MAV calculates the average of absolute values, indicating the activity level of muscles as (10).

$$MAV = \frac{1}{N} \sum_{i=1}^{N} |x_i| \tag{10}$$

Where N is the number of samples and x_i is the *i-th* sample.

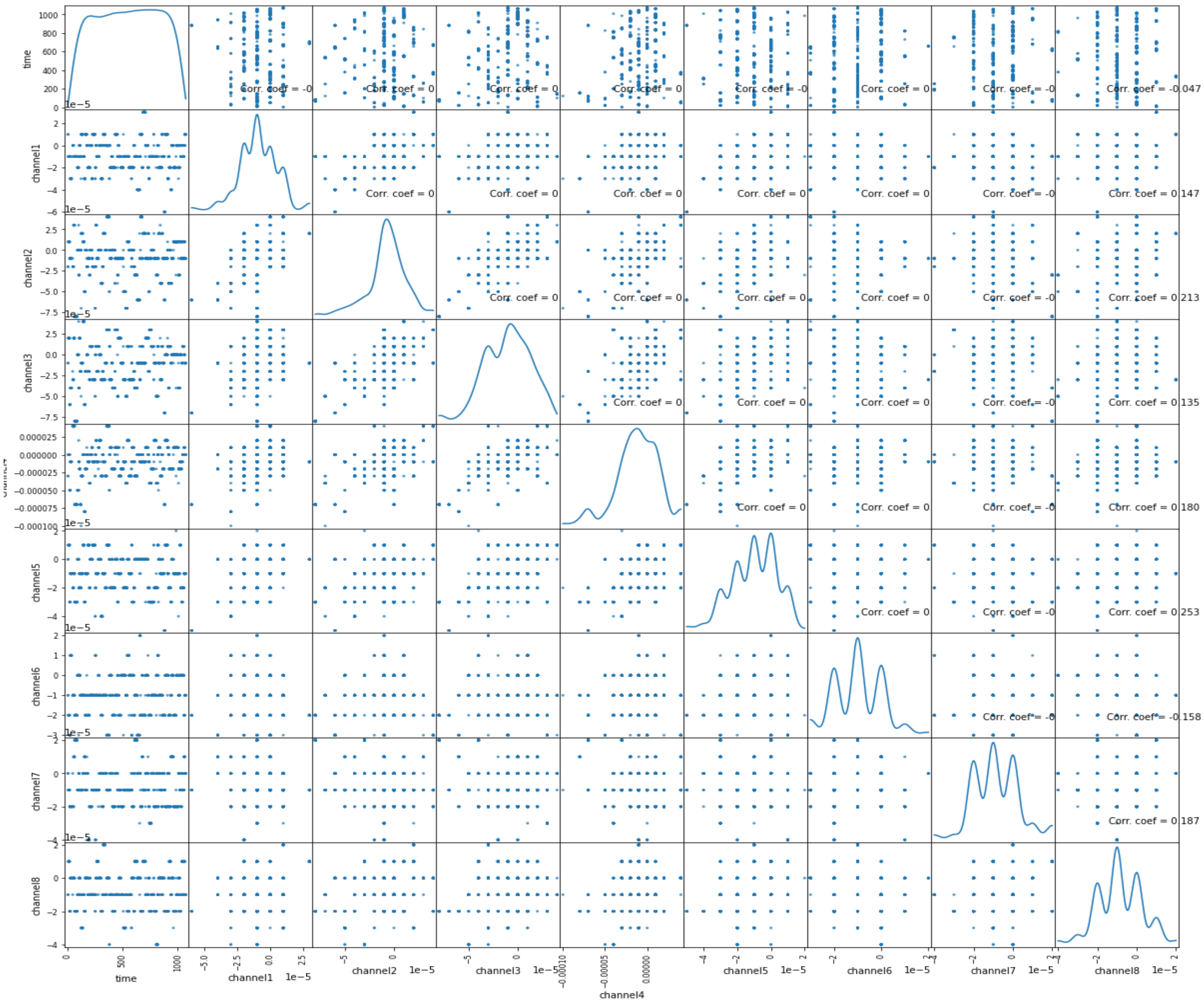

Fig. 5 Scatter and density plot

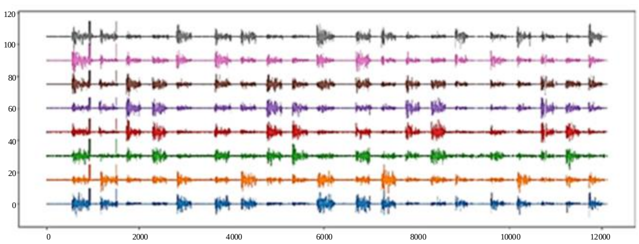

Fig. 6 Output after applying FFT transform

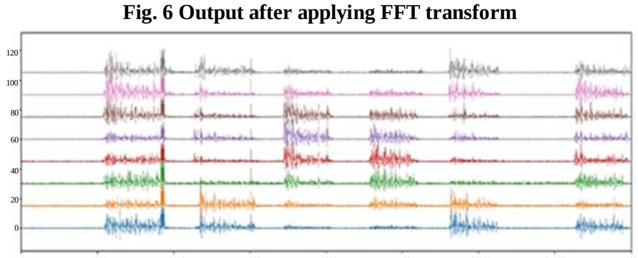

Fig. 7 Output after extracting the envelope

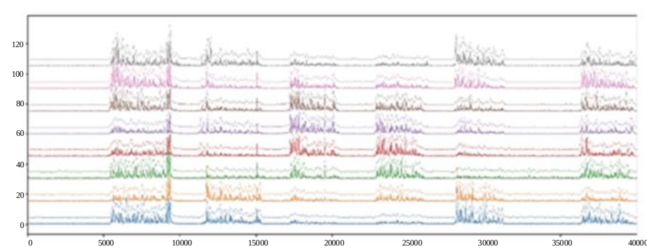

Fig. 8 Output of smoothened envelope

WL measures the cumulative length of the waveform with respect to time, as given by (11).

$$WL = \frac{1}{N}\sum_{i=1}^{N}|x_{i+1} - x_i| \tag{11}$$

ZCR gives the frequency at which the signal changes its sign as (12).

$$ZCR = \frac{1}{N-1}\sum_{i=1}^{N-1} 1_{(x_i \cdot x_{i+1<0})} \qquad (12)$$

The square root of the average of the squares of signal values is given by RMS value as illustrated in (13).

$$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2} \qquad (13)$$

SSI is calculated as the sum of squares of signal values, represented by (14).

$$SSI = \sum_{i=1}^{N} x_i^2 \qquad (14)$$

The Absolute Differences Signal (ABS_DIFFS) is calculated by (15).

$$\text{ABS\_DIFFS} = \sum_{i=1}^{N} |x_{i+1} - x_i| \qquad (15)$$

Figure 9 shows the output of the feature extracted for channel 1. Features with minimum labels are eliminated during feature extraction.

	label	class	channel1								...
			min	max	ptp	SSI	rms	abs_diffs_signal	mav	wavelength	...
0	1	1	-0.00005	0.00004	0.00009	0.000002	0.000017	0.01070	0.000013	0.01070	...
1	1	2	-0.00111	0.00095	0.00206	0.000301	0.000210	0.16819	0.000159	0.16819	...
2	1	3	-0.00087	0.00112	0.00199	0.000260	0.000188	0.14544	0.000138	0.14544	...
3	1	4	-0.00020	0.00016	0.00036	0.000022	0.000056	0.03609	0.000040	0.03609	...
4	1	5	-0.00031	0.00061	0.00092	0.000030	0.000066	0.04939	0.000045	0.04939	...
...	...	...	...	...	...	...	...	...	...	...	...

Fig. 9 Extracted feature from channel 1

The Power Spectral Densities (PSDs) of two signals, the original signal and the smoothened envelope, are employed to analyze the frequency content and assess the impact of the filtering process and the smoothing effects. Figure 10 illustrates the PSD of the original and smoothened version of the signal.

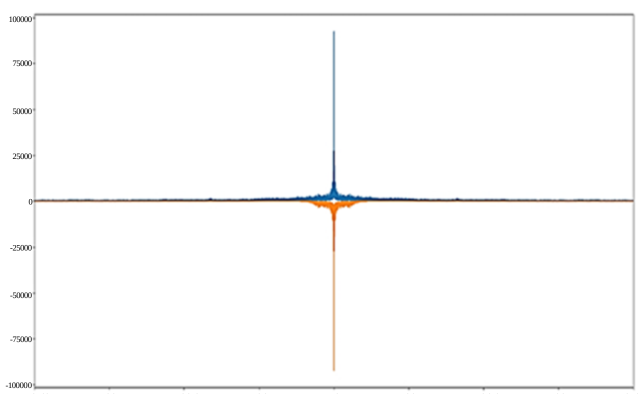

Fig. 10 PSD of the original and smoothened version of the signal

3.4. Proposed Attention-Based LSTM-GRU

The suggested model is designed to identify the hand gestures from the input EMG signals. The model uses a hybrid combination of attention-based LSTM and GRU to predict the intended actions precisely. The input layer processes the input sequence using a self-attention mechanism to extract the more pertinent features for classification. In the process of attention mechanism, the model initially calculates a score for each element of the input sequence and a given query using (16),

$$Scoring\ function, s_i = score(q, h_i) \qquad (16)$$

Where q is the query, and h_i is the input element.

These scores show the significance of each input element regarding the query. Afterwards, the raw attention scores undergo normalization using a softmax function, guaranteeing that the resulting attention weights sum up to one, thus creating a probability distribution across the input elements.

Then, attention weights are generated through element-wise multiplication with the feature maps to highlight significant regions of the input. Mathematically, the attention weight calculation is represented by (17).

$$\alpha_i = \frac{exp(s_i)}{\sum_{j=1}^{N} exp(s_i)} \qquad (17)$$

Where N represents the total number of input elements.

These attention weights are then employed to compute a weighted sum of the input elements, generating an attended representation as in (18).

$$c = \sum_{i=1}^{N} \alpha_i \cdot h_i \qquad (18)$$

As shown in Figure 11, the weighted sum is the content vector representing the most pertinent sections of the input data.

This attention-enhanced representation is fed into LSTM, an RNN architecture for solving the problem of long-term dependencies in the data using a gating mechanism. There are multiple memory gates in each memory cell of the LSTM, including forget input and output gates. LSTM cell is the buried layer of LSTM. LSTM cell architecture is given in Figure 12.

The LSTM gates and state equations are represented by the following:

Cell State:

$$c_t = f_t * c_{t-1} + i_t * \tanh(W_c x_t + W_{ch} h_{t-1} + b_c) \quad (19)$$

Output Gate:

$$O_t = \sigma(W_{ox} x_t + W_{oh} h_{t-1} + b_o) \quad (20)$$

Input Gate:

$$i_t = \sigma(W_{ix} x_t + W_{ih} h_{t-1} + b_i) \quad (21)$$

Hidden State:

$$h_t = O_t * tanh(c_t) \quad (22)$$

Forgot Gate:

$$f_t = \sigma\left(W_{fx} x_t + W_{fh} h_{t-1} + b_f\right) \quad (23)$$

The input vectors are coupled with distinct weight matrices, W_f, W_i, W_c. The symbol σ stands for the sigmoid activation function, while element-wise multiplication is indicated by the *. Furthermore, the bias values for the input cell state forgot the variables specified gate and output gate b_i, b_f, b_c, and b_o, respectively.

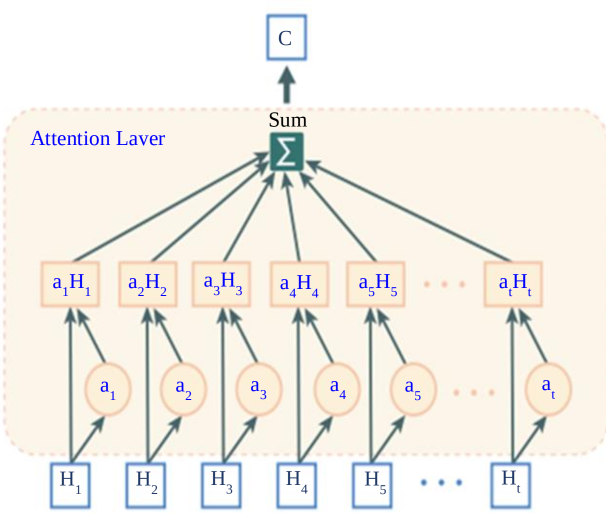

Fig. 11 Illustration of attention mechanism

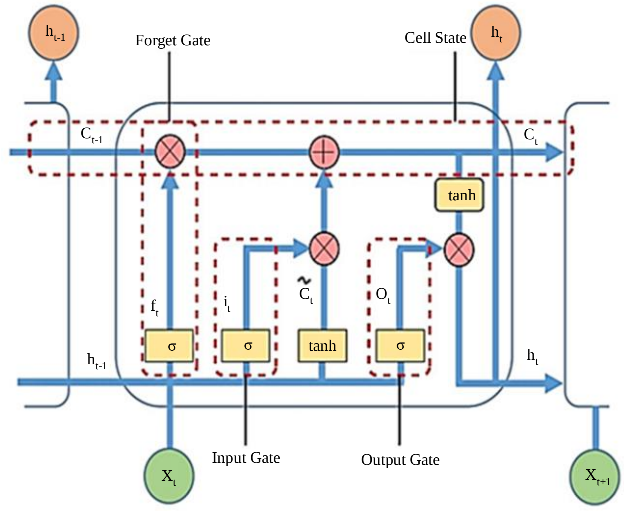

Fig. 12 Basic LSTM cell architecture

The GRU architecture is followed by an LSTM layer to further refine the temporal patterns and dependencies. As shown in Figure 13, the GRU architecture addresses the problem of vanishing gradients by employing gating mechanisms. Unlike traditional RNNs, GRUs utilize a hidden state instead of a separate cell state, generating a new hidden state at each time step by combining input and the previous hidden state.

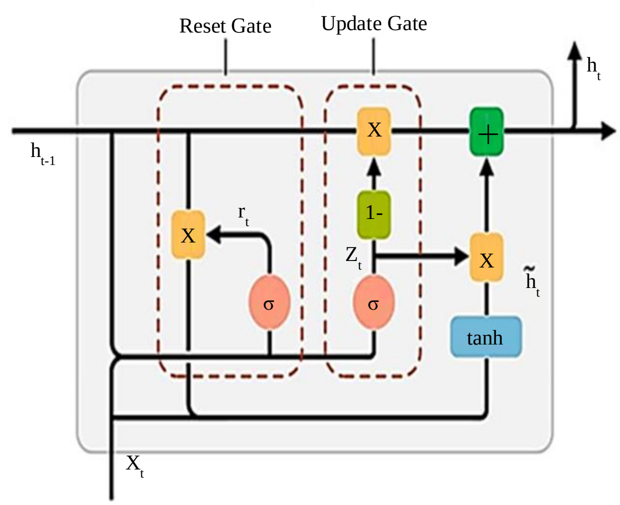

Fig. 13 Basic GRU architecture

The process involves two key gates: the reset gate r_{ti} and the update gate z_{ti}.The reset gate controls the rate at which the previous state is forgotten, as in (24),

$$r_{ti} = \sigma(W_r . [h_{ti-1,} x_{ti}] + b_r) \quad (24)$$

While the update gate manages how much of the prior state is retained, as represented by (25).

$$z_{ti} = \sigma(W_z.[h_{ti-1,}x_{ti}] + b_z) \quad (25)$$

The candidate activation h_{ti} captures new data and integrates it into the hidden state expressed as (26),

$$h_{ti} = (1 - z_{ti}) * h_{ti-1} + z_{ti} * \overline{h_{ti}} \quad (26)$$

Thus, the proposed hybrid model combines LSTM and GRU layers with an attention mechanism to effectively capture spatial and temporal dependencies in EMG signals. The attention mechanism assigns weights to the most significant input features, ensuring the model focuses on critical data segments. The LSTM layers capture long-term dependencies, while GRU layers address vanishing gradient issues, ensuring efficient learning. This combination leverages the strengths of both architectures to improve recognition accuracy. The final output is obtained through a fully connected layer followed by a SoftMax function for gesture classification.

3.5. Hardware and Software Setup

A comprehensive setup is used for this study to ensure a well-equipped environment to handle the demand of neural network training and deployment consisting of NVIDIA GeForce GTX 1080Ti GPU, an Intel Core i7 processor, 32GB of RAM, and the Python-based Keras library integrated with the TensorFlow framework. With Google Colab's vast computing resources and Keras's user-friendly interface, building models was made easy and complex structures were ensured to be trained and executed successfully. Hyperparameters are critical parameters that specify the operation and functions of a deep learning framework throughout the training. Table 2 demonstrates hyperparameters, which are user-specified prior to training, in contrast to the model's parameters, which are determined by the data.

Table 2. Hyperparameter specifications

Hyperparameters	Values
Loss Function	Sparse Categorical Cross Entropy
Activation Function	SoftMax
Dropout	0.5
Batch Size	512
Optimizer	Adam
Epoch	100

4. Results and Discussions

4.1. Performance Evaluation

Performance evaluation of the model was conducted to ensure a comprehensive understanding of its effectiveness using a variety of metrics. The primary metrics, shown in Table 3, offers insights into the model's performance. These metrics were chosen for their relevance in tasks requiring detection accuracy and minimizing false positives/negatives.

Table 3. Evaluation parameters

Performance Metrics	Equations
Accuracy	(TP+TN)/(TP+TN+FP+FN)
Precision	TP/(TP+FP)
Recall	TP/(TP+FN)
F1 Score	2*(Precision*Recall) / (Precision+Recall)
Where, TP-True Positives, FP-False Positives, TN-True Negatives, and FN-False Negatives	

With the help of the evaluation metrics, the effectiveness of the hybrid model for HGR is evaluated. The classification report of the suggested model is illustrated in Table 4, and Figure 14 represents its visualization.

Table 4. Classification report of proposed method

Evaluation Metrics	Results (%)
Accuracy	0.9887
Precision	0.9782
Recall	0.9807
F1- Score	0.9774

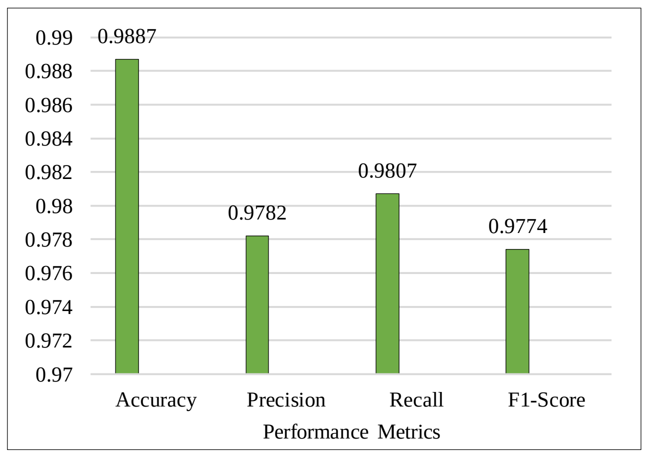

Fig. 14 Graphical representation of performance evaluation

Table 4 shows that the hybrid Attention-based LSTM-GRU is highly effective in recognizing hand gestures. The model performs well with an accuracy of 98.875%, indicating the correctness in recognizing most hand gestures. The model shows 97.82% precision, indicating the effectiveness in predicting particular gestures, reflecting a low false positive rate. With a recall rate of 98.07%, the model accurately detects the real motions, demonstrating a low false negative rate. F1-Score of 97.74%, a combined score of precision and recall, demonstrates a balanced performance in both gesture recognition and accurate classification.

The study's accuracy and loss plots were crucial for assessing the model's performance throughout training. The accuracy plot demonstrated the model's learning progress, which showed strong generalization as training accuracy increased gradually and validation accuracy followed suit. The consistent decay in the loss plot indicates that the model has fewer errors and is more efficient in learning. No divergence between the training and validation metrics indicates no overfitting. The model's accuracy and loss plot are shown in Figure 15. The confusion matrix depicted in Figure 16 illustrates the model performance by comparing it with the actual labels. Predicted labels are compared to actual labels, with accurate predictions along the diagonal and incorrect classifications shown by off-diagonal parts. Several performance metrics are calculated from the confusion matrix to assess the framework's efficiency, as indicated in Table 3.

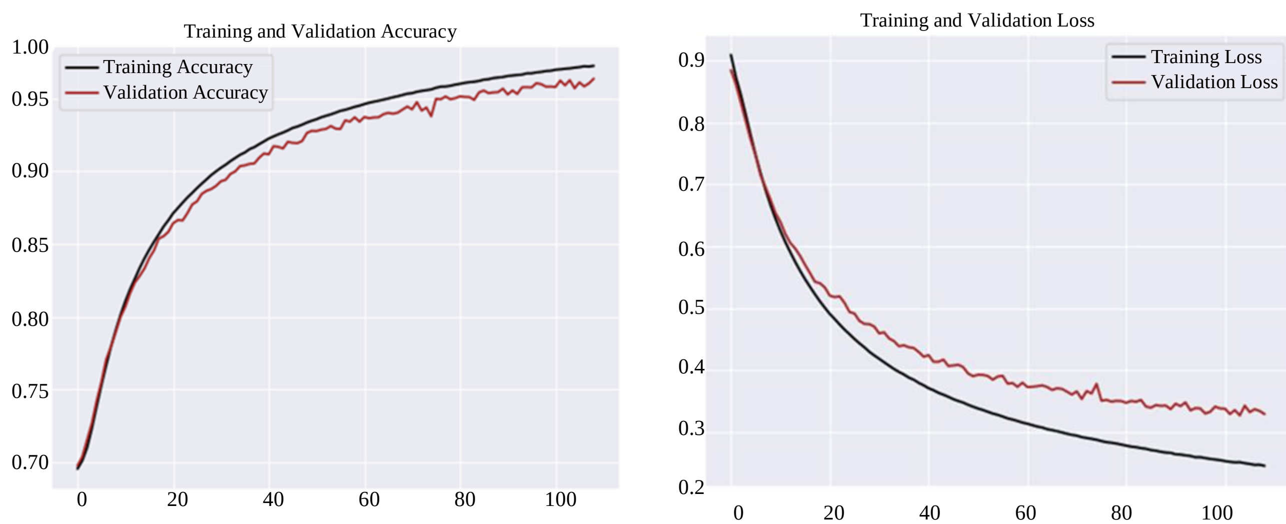

Fig. 15 Accuracy and loss plot of the suggested hybrid model

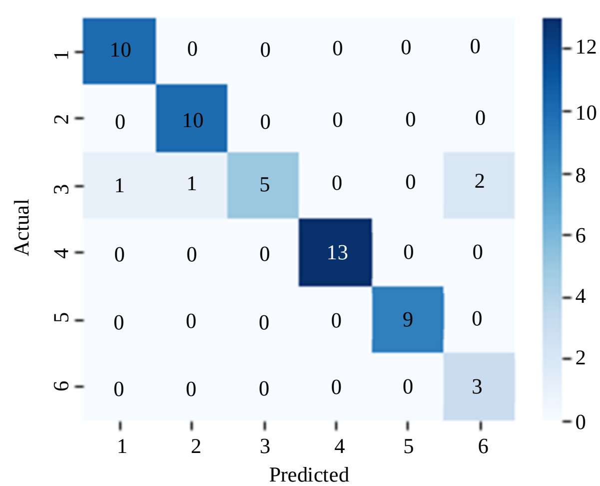

Fig. 16 Confusion matrix of the suggested model

4.2. Performance Comparison

Table 5 and Figure 17 show the proposed model's effectiveness compared to existing models. The effectiveness of several models has been demonstrated in Table 5. The LSTM model offers an accuracy of 0.818 and reasonably better performance in capturing temporal dependencies, and CNN achieves an accuracy of 0.850 by effectively capturing spatial information from EMG signals. By combining spatial and temporal feature extraction, the hybrid CNN+LSTM model provides an accuracy of 0.921, showing a considerable performance improvement.

The proposed Attention-based LSTM-GRU model with an accuracy of 0.9887, precision of 0.9782, recall of 0.9807, and F1 score of 0.9774 is more effective in hand gesture recognition by analyzing complex patterns from the EMG signals. Figure 17 shows the accuracy of comparing the suggested model with the existing methods.

Table 5. Comparison with the state of the art model

Model	Accuracy	Precision	Recall	F1 Score
LSTM	0.818	0.792	0.800	0.795
BiLSTM	0.728	0.730	0.76	0.744
GRU	0.835	0.802	0.831	0.815
CNN	0.850	0.832	0.826	0.829
CNN+LSTM	0. 921	0.900	0.878	0.912
Proposed model	**0.9887**	**0.9782**	**0.9807**	**0.9774**

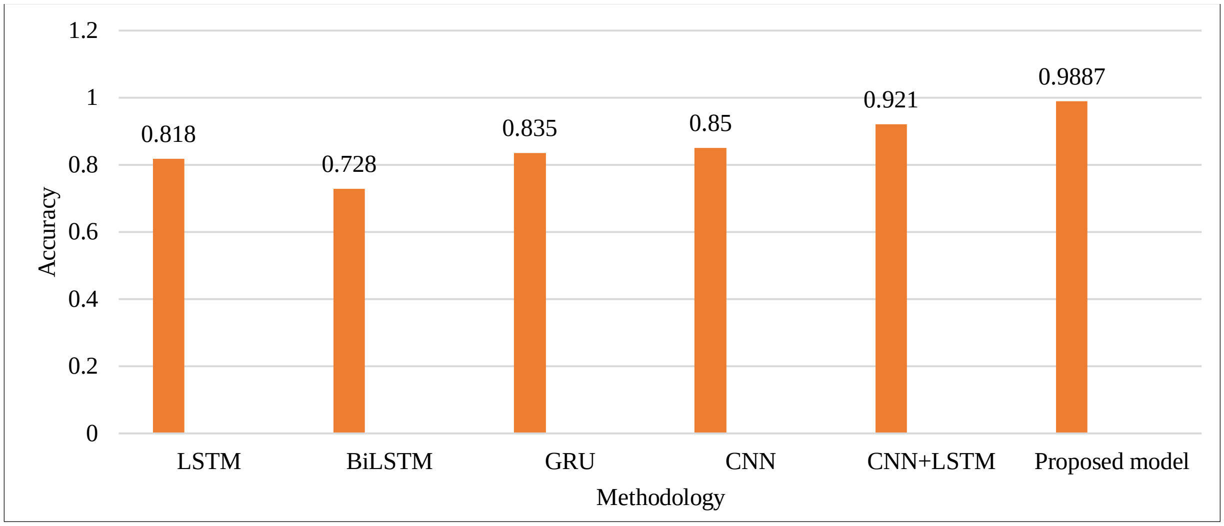

Fig. 17 Accuracy comparison of the suggested model with the existing methods

5. Conclusion

HGR improves human-computer interaction by facilitating natural communication with gadgets and offering crucial accessibility for those with disabilities. This research suggested a hybrid deep learning model combining attention-based LSTM and GRU networks for efficient HGR using EMG signals. The suggested model utilizes the capability of both LSTM and GRU to capture the temporal dependencies. An attention mechanism is incorporated to extract the pertinent features of the data. Frequency domain and time domain analysis are evaluated for efficient gesture recognition. The study demonstrates that the hybrid model achieves superior performance metrics, with 98.875% accuracy, 97.82% precision, 98.07% recall, and an F1 score of 97.74%, significantly outperforming existing methods.

These findings demonstrated the robustness and the model's potential for real-time applications, including assistance for individuals with disabilities and advanced HCI, resulting in enhanced quality of life. While the model demonstrates strong performance, challenges like noise in EMG signals, dataset limitations (static gestures only), and computational demands should be acknowledged.

Acknowledgments

I want to express my sincere gratitude to all those who contributed to completing this research paper. I extend my heartfelt thanks to my supervisor, family, colleagues and fellow researchers for their encouragement and understanding during the demanding phases of this work.

References

[1] Munir Oudah, Ali Al-Naji, and Javaan Chahl, "Hand Gesture Recognition Based on Computer Vision: A Review of Techniques," *Journal of Imaging*, vol. 6, no. 8, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[2] Okan Köpüklü et al., "Real-Time Hand Gesture Detection and Classification Using Convolutional Neural Networks," *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition*, Lille, France, pp. 1-8, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[3] Arpita Ray Sarkar, G. Sanyal, and S. Majumder, "Hand Gesture Recognition Systems: A Survey," *International Journal of Computer Applications*, vol. 71, no. 15, pp. 26-37, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[4] Nurettin Çağrı Kılıboz, and Uğur Güdükbay, "A Hand Gesture Recognition Technique for Human-Computer Interaction," *Journal of Visual Communication and Image Representation*, vol. 28, pp. 97-104, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[5] William T. Freeman, and Michal Roth, "Orientation Histograms for Hand Gesture Recognition," *International Workshop on Automatic Face and Gesture Recognition*, vol. 12, pp. 296-301, 1995. [Google Scholar] [Publisher Link]

[6] Swapnil D. Badgujar et al., "Hand Gesture Recognition System," *International Journal of Scientific and Research Publications*, vol. 4, no. 2, pp. 1-5, 2014. [Google Scholar] [Publisher Link]

[7] Xu Zhang et al., "A Framework for Hand Gesture Recognition Based on Accelerometer and EMG Sensors," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 41, no. 6, pp. 1064-1076, 2011. [CrossRef] [Google Scholar] [Publisher Link]

[8] Enea Ceolini et al., "Hand-Gesture Recognition Based on EMG and Event-Based Camera Sensor Fusion: A Benchmark in Neuromorphic Computing," *Frontiers in Neuroscience*, vol. 14, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[9] Lorena Isabel Barona López et al., "CNN-LSTM and Post-Processing for EMG-Based Hand Gesture Recognition," *Intelligent Systems with Applications*, vol. 22, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[10] Mina Pourmokhtari, and Borhan Beigzadeh, "Simple Recognition of Hand Gestures Using Single-Channel EMG Signals," *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, vol. 238, no. 3, pp. 372-380, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[11] Yuxuan Wang et al., "A Hand Gesture Recognition Strategy Based on Virtual-Dimension Increase of EMG," *Cyborg and Bionic Systems*, vol. 5, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[12] Mai H. Abdelaziz, Wael A. Mohamed, and Ayman S. Selmy, "Hand Gesture Recognition Based on Electromyography Signals and Deep Learning Techniques," *Journal of Advances in Information Technology*, vol. 15, no. 2, pp. 255-263, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[13] Zhongpeng Zhang et al., "Hardware and Software Design and Implementation of Surface-EMG-Based Gesture Recognition and Control System," *Electronics*, vol. 13, no. 2, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[14] Hassan Ashraf et al., "Optimizing the Performance of Convolutional Neural Network for Enhanced Gesture Recognition Using sEMG," *Scientific Reports*, vol. 14, no. 1, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[15] Ahmed M. Al-Khazzar, Zainab Altaweel, and Jabbar S. Hussain, "Using Deep Neural Networks in Classifying Electromyography Signals for Hand Gestures," *International Journal of Artificial Intelligence*, vol. 13, no. 1, pp. 217-227, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[16] Ulysse Côté-Allard et al., "Deep Learning for Electromyographic Hand Gesture Signal Classification Using Transfer Learning," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 4, pp. 760-771, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[17] Reema Jain, and Vijay Kumar Garg, "EMG Signal Feature Extraction, Normalization and Classification for Pain and Normal Muscles Using Genetic Algorithm and Support Vector Machine," *Revue d'Intelligence Artificielle*, vol. 34, no. 5, pp. 653-661, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[18] Alejandro Mora Rubio et al., "Identification of Hand Movements from Electromyographic Signals Using Machine Learning," *Preprint.org*, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[19] N. Krilova et al., EMG Data for Gestures, UCI Machine Learning Repository. [Online]. Available: https://archive.ics.uci.edu/dataset/481/emg+data+for+gestures

[20] Gözde Karataş Baydoğmuş, "The Effects of Normalization and Standardization an Internet of Things Attack Detection," *Avrupa Bilim ve Teknoloji Dergisi*, no. 29, pp. 187-192, 2021. [CrossRef] [Google Scholar] [Publisher Link]