Al-Rafidain Engineering Journal (AREJ) Vol.30, No.1, March 2025, pp. 150-163

Advancing EMG Finger Movement Classification with Feature Extraction and Machine Learning

Rasha A. Moyassar rasha.moysar2014@stu.uoninevah.edu.iq
Mohammed A. M. Abdullah mohammmed.abdulmuttaleb@uoninevah.edu.iq

Computer and Information Engineering Department, College of Electronics Engineering, Ninevah University, Mosul, Iraq

Received: August 5th, 2024 Received in revised form: October 7th, 2024 Accepted: October 27th, 2024

ABSTRACT

In prosthetic finger development using electromyogram (EMG) data, a crucial challenge is accurately recognizing finger movements, thus requiring developed models that process EMG signals, facilitating independent finger gesture classification with high accuracy. To successfully classify an EMG signal, the feature selection should be carefully evaluated. However, many studies on EMG signal classification have employed a feature set that includes several redundant elements. In this study, several combinations of time domain features are employed for EMG signal reduction. In addition, two models of CNN namely: (CNN-1, CNN-2), DFNN, LSTM, and GRU architectures are proposed to provide high accuracy with minimal computational overhead and minimum parameters. Through careful model selection and hyperparameter optimization, the models' effectiveness was enhanced. The models were evaluated based on accuracy, precision, recall, and F1-score metrics. Among the proposed models, CNN-1 resulted in a good balance in terms of accuracy, computational time, and memory size, with an accuracy of 97.3 in 0.96 minutes with 890.73 KB size of memory. Furthermore, a comparison against earlier work confirmed the efficacy of the approach.

Keywords:
Electromyography (EMG), Machine Learning (ML), Convolution Neural Network (CNN), Deep Feedforward Neural Network (DFNN), Long Short-Term Memory (LSTM), Gated Recurrent Units (GRUs).

https://rengj.mosuljournals.com
Email: alrafidainengjournal3@uomosul.edu.iq

===

1. INTRODUCTION

The essential components of our body are muscles, enabling a wide range of actions. Via muscle contraction and relaxation, produced myoelectric signals, allowing movement. These signals can be obtained using electrodes of EMG signals, to allow for the emulation of human movements, and are applicable in different fields [1]. EMG performs an essential function in clinical environments by monitoring muscle activity, providing neuromuscular disorders, and directing rehabilitation efforts. Moreover, the connection of EMG to technological fields and robotics has opened avenues for its application in areas such as human-computer interfaces, prosthetic development, gaming, and assistive technologies, demonstrating its versatility across multiple interdisciplinary fields [2]. Typically, collecting EMG data uses either surface electrodes, which are put on the skin, or embedded electrodes, which are implemented directly into the muscle tissue. Each type of electrode provides useful insights into muscle function and movement [3]. Indeed, surface EMG shows a significant noninvasive measure of muscle activity, which can be incredibly useful in robotic applications for controlling artificial organs [4]. EMG signals classification utilizing dependable and robust approaches is becoming increasingly popular in the field of biomedical engineering [5]. Three major cascaded modules must be meticulously managed to effectively classify and recognize EMG signals: data pre-processing, feature extraction, and classification algorithms, with a focus on selecting the best feature vector. Feature extraction is critical in revealing useful information concealed within surface EMG signals (s EMG) while also removing useful elements and interferences. This step is necessary for improving the efficiency and accuracy of the subsequent classification stages [6]. Classification of EMG is important in interpreting wrist and elbow movements, grip, and finger movements, with considerable accuracy. Despite these significant advancements, there is still a gap between laboratory-based results and actual clinical applications, which requires additional efforts to close. Ensuring that these breakthroughs can be efficiently and reliably applied in real-world clinical settings requires resolving some problems such as the durability of technology in varied surroundings, unpredictability in patient physiology, and user-specific customization [7]. The Advancement of machine learning approaches increases the ability to improve the precision and versatility of EMG-based classifications. Among the most common computational models in Artificial Intelligence (AI) employed for this purpose are Random Forest, k-nearest Neighbors (KNN), and deep learning approaches, particularly Artificial Neural Networks (ANN) and Convolution Neural Networks (CNNs).

In a wide range of applications, Long Short-Term Memory Neural Networks (LSTMs) have recently had promising results. Recurrent Neural Networks (RNNs) specifically engineered for time series analysis, including EMG signals LSTMs are a type of RNN and address the vanishing gradient problem in classic recurrent neural networks, allowing for the effective capture of long-term dependencies. The gated architecture enables either retaining or forgetting information over time, making it ideal for replicating complicated temporal patterns found in EMG data. Over the years, LSTMs have proven their ability to learn from sequential data and extract essential features making them particularly powerful for applications like EMG signal categorization, which requires understanding the temporal dynamics of muscle activation and outstanding performance in a variety of disciplines. Experiments show that the LSTM prediction algorithm outperforms numerous traditional time series prediction algorithms [8].

Similarly, an RNN version called Gated Recurrent Units (GRUs) aims to prevent gradient vanishing, just as LSTM. In contrast to LSTM, GRU integrates input and forget gates into a signal update gate to optimize cell structure. This lowers processing complexity and, in certain cases, results in performance that is on par with or even better than LSTM for shorter training times. [9]. This adaptability highlights their relevance across multiple domains, making them valuable instruments for investigators seeking dependable and efficient time series analysis resolutions. For instance, GRU has several advantages to using efficiently and reliably applied in EMG signal categorization, including accuracy, responsiveness, and computing economy.

1.1 Employed ML Algorithms Background

1.1.1. Gradient Boosting

Gradient Boosting (GB), is a supervised learning technique. Boosting is a type of ensemble algorithm where predictors are developed in sequence, not in isolation. In this method, each subsequent tree builds upon a revised version of the initial dataset. Boosting methods amalgamate the outputs from these sequential models using a method of weighted averaging to arrive at the final decision [10].

1.1.2. Random Forest

A random forest is an ensemble of decision tree classifiers combined to create a collective model. Each decision tree within the forest is uniquely structured, as the decision splits at each node are made by selecting from a random subset of attributes. This process ensures that each tree develops distinct decision paths. Furthermore, each tree in the forest is associated with the same distribution, but they differ in the specifics because the values of the random vector used to choose the attributes and data samples are unique for each tree. The creation of random forests involves the techniques of bagging (bootstrap aggregating) and random attribute selection, enhancing the model's ability to generalize across different data samples and reducing the likelihood of overfitting [11].

1.1.3. Support vector machines

SVM, a type of machine learning method rooted in statistical learning theory, is widely used in pattern recognition. Its notable features include the absence of local minima, sparsity in solutions, and the utilization of kernel-induced feature spaces [12]. While most prior classifiers rely on hyperplanes to separate classes, SVM extends this concept to data that cannot be linearly separated by mapping predictors to a new, higher dimensional space where linear separation is possible. Typically, misclassifications occur only when an inappropriate kernel function is chosen or in cases with significantly different classes. From a computational standpoint, determining the optimal position for the decision plan becomes an optimization problem, aiding in the creation of linear boundaries through non-linear transformations [13].

1.1.4. Deep Feedforward Neural Network

A feedforward neural network (FFNN) is a basic type of artificial neural network characterized by unidirectional information flow. Data moves linearly from the input layer, through hidden layers, to the output layer without cycles. In each layer Neurons are equipped with activation functions, applying weights to inputs by process data, allowing the network to learn complex patterns during training [14]. Compared to traditional FFNNs, Deep Feedforward Neural Network (DFNN) has a larger number of hidden layers. This feature allowed DFNN to perform more complicated tasks by automatically learning and representing hierarchical features from input data.

1.1.5. Convolution Neural Network

Due to their strong data processing capabilities Convolutional Neural Networks or CNNs are widely employed in machine learning for tasks such as natural language processing and computer vision. CNNs contain convolution and pooling layers. Convolution layers use convolution kernels to interact with specific areas of input data, and their parameters change throughout training. Transform outputs using non-linear activation functions such as Rectified Linear Unit (ReLU) and hyperbolic tangent. After convolution layers come Pooling layers, which reduce dimensionality and summarize features using techniques such as max-pooling. CNN architectures often have various convolution and pooling layers, then fully connected layers, enabling capturing hierarchical patterns necessary for complicated recognition tasks and efficient data processing [15].

1.1.6. Recurrent Neural Network

Another type of FFNN is a Recurrent Neural Network (RNN) that includes loops into its hidden layers. In the input, these loops give the model to process sequences of data while capturing temporal relationships. Additionally, Long Short-Term Memory (LSTM) networks address the problem that classic RNNs frequently struggle to understand long-term dependencies due to difficulties such as vanishing gradients by including techniques for modulating information flow in the hidden layer loops. This architecture enables LSTM networks to forget or keep states based on their relevance to the task, giving the network a form of long-term memory. As a result, in learning from sequence data, LSTM is generally more effective than traditional RNNs [14]. Gated Recurrent Units (GRUs) are another adaptation that simplifies the LSTM design by combining the forget and input gates into a signal update gate, as well as merging the cell state and hidden state, thereby streamlining the architecture and making it easier to train while still effectively capturing long-term dependencies. As a result, in general, LSTMs and GRUs outperform standard RNNs when learning from sequence data.

This paper is organized as follows. Section 2, presents the related work. In Section 3, the proposed method is presented. Section 4 illustrates the results and comparisons with similar work. Finally, Section 5 concludes this paper.

2. RELATED WORKS

Several researchers have developed multiple algorithms for feature extraction and pattern recognition to enhance the decoding of signals and achieve precise finger classification. For instance, Srinivasan et al [16] outlined how different finger flexions at rest are analyzed using EMG signals. Five different actions, including thumb flexion, index flexion, ring flexion, little flexion, and the rest are classified using the CNN model. Ten subjects were used to gather data and signal processing techniques with underwent preliminary processing to enhance signal clarity and filter out noise. This custom dataset demonstrates its potential for effective EMG signal classification achieved results of an accuracy of 72.5. Naseer et al [17] employed deep neural networks, LDA, SVM, and KNN to categorize EMG signals from five specific fingers across ten participants. using an eight-channel. Features are extracted from these signals via processing and the accuracy varied from 92.7 to 97.4. Bhattachargee et al [10] introduced a novel method designed to differentiate between EMG signals generated from ten distinct hand gestures and eight participants using the Gradient Boosting (GB) classifier. Both statistical and frequency features were extracted from the raw EMG data to simplify the signals and enhance their interpretability for the classification algorithm, it was applied to a practical EMG dataset to validate the effectiveness of this method, achieving 98.5 accuracy. Krishnan et al [18] proposed and evaluated various algorithms for classifying finger movements using EMG sensors, focused on eight distinct finger motions. For this classification task, two classifiers: Linear Discriminant Analysis (LDA) and SVM are utilized, each tested with different features. From the analysis of the results, it was found that LDA achieved the highest classification accuracy at 97.7. However, a notable drawback of LDA is its limited tractability for real-time applications. Conversely, SVM provided a more favorable balance between speed and accuracy, achieving an accuracy of 95.7. Tepe et al [1] employed a model that recognizes finger gestures by processing EMG signals. During preprocessing, the dataset of five distinct finger and resting hand gestures underwent filtering to identify segments where gestures occurred, followed by a windowing process. A classification rate of 95.8 was achieved utilizing the Simplified KNN method with the extracted waveform length feature within a 100 ms window and 50 overlaps. Fındık et al [19] automatic creation and selection of EMG signal features were proposed, then developed a classification method based on a Random Forest algorithm. Ten distinct finger motions were recorded by two EMG sensors, which achieved an accuracy of 97.5. Arteaga et al [20] proposed a primary approach for robot-assisted hand motion therapies with two initial objectives. Time and frequency features are used for recognizing six hand gestures as inputs for multiple algorithms of machine learning. Specifically, compared the performance of ANN, Support Vector Machines (SVM), and KNN algorithms for classification effectiveness. Interpolation methods are used for each identified gesture to be transformed into a joint reference trajectory. The average correlation between actual tracked hand motion and EMG-based generated joint trajectories was notably high at 0.91. Moreover, statistical analysis of 14 different configurations of ANN, SVM, and KNN algorithms revealed that KNN and Weighted KNN algorithms performed well achieving the best classification accuracy of 98. Millar et al [12] employed neural networks with feature extraction, to achieve higher accuracy in classification. An LSTM network has been developed for this purpose and has successfully classified 12 distinct finger movements with an accuracy of 90. Taghizadeh et al [4] utilized the advanced Fractional Fourier Transform (FrFT) technology to extract features from the EMG of eight subjects. Ten different finger movements were used to record these signals. Both the windowing and t-test approaches were used to optimize feature selection. KNN algorithm was used to classify these features, achieving an average classification accuracy of 98.12. Lee et al [21] presented a classifier for EMG-based hand and finger gestures. Ten healthy participants performed ten different hand and finger gestures collected from EMG each channel extracted features with a six-Time Domain (TD). These features were used to generate individual classifiers for each gesture using SVM, Random Forest (RF), ANN, and Logistic Regression (LR). This approach achieved results showing that ANN has the greatest mean accuracy at 0.940, followed by the SVM at 0.876, the RF at 0.831, and the LR at 0.539. Tepe et al [22] enhanced classification accuracy, by achieving a reliable classification of sEMG data from finger movements. The authors incorporated gyroscopic signals, into the sEMG data analysis. Ten normal subjects were collected data, each performing 6 finger gestures thirty times. The EMG signals were preprocessed to extract features, and then the sequential forward feature selection method to identify the most effective feature set for classification. For classification, SVM, KNN, and multilayer ANN algorithms are utilized. Findings indicated that using features extracted solely from sEMG data, the ANN method achieved the best performance, with an accuracy of 94.40. When features from both sEMG and gyroscopic data were used, the performance improved to 96.30 with a significant p-value (<0.05). Kumar et al [23] described a Machine Learning (ML) framework designed to detect finger movements for a prosthetic hand. Given that the data generated from different movements across various fingers constitutes a multiclass classification problem, employed four ML-based classifiers: KNN, decision tree, RF, and eXtreme Gradient Boosting (XGBoost). Experiments demonstrate that the XGBoost classifier surpassed other classifiers in terms of accuracy, making it the most effective tool for classifying complex finger movement within the context of prosthetic hand control. Sultana et al [24] focused on analyzing the accuracy, applicability, and efficiency of different machine learning algorithms for hand and finger gesture recognition. Sultana et al [25] disclosed a new method for analyzing and classifying 15 different finger movements from eight healthy individuals using surface EMG signals. This technique uses Welch power estimate for frequency analysis to enhance the classification process. Five TD features were extracted from these signals and employed to construct a machine-learning classifier. The preliminary experimental results demonstrated a classification accuracy of 92.30 for data obtained from eight channels. By concentrating on two prominent channels, this accuracy was raised to 94.15 During the categorization process, employed ten-fold cross-validation to provide a trustworthy performance evaluation. Finally, 25 of the data points were set aside as test data, and the approach achieved an average accuracy of 92.35

3. PROPOSED FRAMEWORK

This section briefly presents employed machine learning algorithms and the proposed methods.

3.1 Time Domain Feature Extraction

TD features are adopted for the identification of finger recognition for this study. Time domain features are extracted directly from raw EMG signals as a function of time, without requiring any transformation. These features can be efficiently computed from the sampled time series, making the process faster and easier to implement, with a reduced computational burden. Common TD features include Mean Absolute Value (MAV), Root Mean Square (RMS), Zero Crossing (ZC), Standard Deviation (SD), minimum, maximum, Waveform Length (WL), Amplitude First Burst (AFB) and Willison AMPlitude (WAMP). These features capture various aspects of the signal’s amplitude, variability, and complexity directly from the time series data [24].

3.2 Features Extraction

Directly inputting raw EMG signals as time sequences into a classifier can lead to decreased classification performance. This is because raw EMG signals encompass extensive sequences with high variability, randomness, and redundancy, which add complexity to the data [26]. Feature extraction is a technique that converts raw signals into a condensed set of attributes known as feature vectors. When appropriately chosen, these feature vectors retain essential and relevant information from the raw EMG signals, effectively representing the intended actions [24]. Generally, features are extracted using the Time Domain (TD), Frequency Domain (FD), or Time-Frequency Domain (TFD). The Time Domain is particularly favored because it is fast, easy to implement, and effectively represents the transient states of gestures [21].

In this study, numerous temporal domain aspects of the EMG signal were analyzed using deep learning algorithms such as CNN, DFNN, LSTM, and GRU classifiers. By combining features, we aim to improve the precision, correctness, and dependability of the outcomes. This approach enables us to improve the effectiveness of our classification system, resulting in more consistent results when applying EMG signal analysis. The results indicated the improved capabilities of our algorithms demonstrating their efficacy in EMG signal analysis.

3.3 Proposed Methodology

In this work, two proposed CNN models namely: CNN-1, and CNN-2, in addition to DFNN, LSTM, and GRU are used for EMG signal classification. Furthermore, Random Forest, SVM, and XGB are utilized to determine which feature achieved the best results to be used in the proposed algorithms. The features were SD, RMS, minimum, maximum, ZC, AAC, AFB, MAV, WL and WAMP. Different combination of features is selected to obtain the highest accuracy. Table 4 summarizes these features.

3.3.1 Employed ML models

As mentioned before, three machine learning models, namely Random Forest, SVM, and XGB, are used for EMG classification, with their hyperparameters tuned empirically to achieve optimal performance. The best hyperparameters for each ML algorithm are outlined in Table 1. A comparative analysis of these algorithms is conducted subsequently. The primary objective of using ML is to determine which features yield higher accuracy due to ML algorithms offering constant accuracy compared to DL algorithms.

Table 1: Show the best hyperparameters of ML algorithms

Model	Hyperparameters
Random Forest	n_estimators = 22 criterion =′ entropy′ random_state = 0 max_depth = 62
SVM	C = 100 gamma = 0.01 decision_function_shape =′ ovo′ kernel =′ poly′
XGB	– –

3.3.2 CNN-1, CNN-2, DFNN, LSTM and GRU models

In this study, we introduce several lightweight neural network models. The first model called CNN-1 is composed of two convolution layers with 16 and 32 filters, respectively, each having a kernel size of three. This is followed by a single max pooling layer with a pool size of two. The architecture also includes two dense layers, each equipped with 64 neurons. The Rectified Linear Unit (ReLU) is used as the activation function. To minimize the risk of overfitting, a dropout rate of 0.5 is implemented. The final layer of the CNN model is a dense output layer with seven neurons, representing the number of classes, and employs a SoftMax activation function, Figure 3, a.

The second proposed model (CNN-2) includes early stopping to prevent overfitting and enhance generalization to new, previously unknown data, as well as two convolutions' layers with 50 and 100 filters, respectively, each with a kernel size of three. To decrease over-fitting, the first convolution layer is followed by a max pooling layer with a pool size of two and a dropout layer with a 0.2 rate. Followed by the convolution layer with a max-pooling of 128 neurons activated by ReLU. The last layer is dense, containing seven neurons, and relates to the seven classes and uses the SoftMax activation function, Figure 3, b. A new DFNN is the third model comprising 2 dense layers, each with 128 neurons, also using the ReLU activation function. A dropout layer of 0.5 rate is recommended to prevent overfitting. The output layer of this model includes seven neurons (indicative number of classes) with SoftMax activation function, Figure 3, c. On the other hand, the suggested GRU model includes two GRU layers. It has one dense layer of 128 neurons and a dropout rate of 0.5 to prevent overfitting. The final layer has a dense layer with 7 neurons, that uses a SoftMax activation function for class prediction, Figure 3, d. Furthermore, our suggested Long Short-Term Memory (LSTM) model features have two LSTM layers with 64 and 60 neurons. To address overfitting, this model incorporates a dropout rate of 0.5. It includes a dense layer with 128 neurons and ends in an output layer with 7 neurons, each corresponding to a class, using a SoftMax activation function Figure 3, e.

Table 3 summarizes the hyperparameters of the employed CNN-1, CNN-2, DFNN, GRU, and LSTM models.

The database used in this work has a small volume. To address this problem, the random over-sampling method is used in DL algorithms. In this method, new samples are generated in the underrepresented classes by randomly sampling the currently available signals.

4. RESULTS AND DISCUSSION

4.1 Dataset

In this work, we employed an EMG signal database which is obtained from Kaggle platform. The data are collected using the MYO Thalamic bracelet, Figure 1. The data was gathered from 10 subjects across 20 trials, employing three different wrist positions on the widest section of the forearm. Using MATLAB integrated with MYO SDK, the collection process involved subjects performing gestures with a one-second pause, repeated for 20 to 30 trials [17], table 2 shows these details. This dataset has finger movement of an index finger, middle finger, ring finger, little finger, thumb, rest, and victory gestures Figure 2. Eight electrodes were used and 10 features were extracted for each electrode. The features in this dataset are SD, RMS, minimum, maximum, ZC, AAC, AFB, MAV, WL, and WAMP. Figure 4 displays the raw data and features extraction respectively.

Table 2: Dataset information

Specification	Dataset
Number of subjects	10
Number of classes	7
Number of features	10
Number of electrodes	8
Number of repetitions	20-30
Time of performing	1-second pause

Fig. 1: The MYO EMG Armband [27]

4.2. Experimental Results

All experiments are implemented using Python programming language in the Kaggle notebook to implement deep learning and machine learning algorithms with GPU P100. As mentioned earlier, we used an EMG dataset obtained from Kaggle[28]. This study assessed the effectiveness of several machine learning algorithms, namely XGB, RF, and SVM, using a progressively complex feature set derived from signal processing. The results of the classification process using a combination of parameters were much higher than the use of individual parameters. The combination of SD RMS + minimum + maximum + ZC + AAC + AFB + MAV + WL yielded the highest results using XGB and RF with an accuracy of 96.2 and 95.9, respectively. Whereas SVM achieves the highest accuracy of 95.7 when combining SD, RMS, minimum, maximum, ZC, AAC, AFB, MAV, WL, and WAMP. Table 4 shows these results. This study also evaluated the performance of several neural network models on a classification task, assessing their accuracy, computational efficiency (measured in execution time), memory requirements (total parameters), and optimizer parameters in combination with SD, RMS, minimum, maximum, ZC, AAC, AFB, MAV and WL. The goal was to identify which model architecture offers the best balance of high accuracy and operational efficiency for real-time applications, including prosthetic limb control and gesture recognition systems, where precise muscle activity classification is essential.

As shown in Table 5, and Figure 5, the proposed CNN-1 achieves a higher accuracy of approximately 97.3. Additionally, satisfactory accuracy, loss curves, and a confusion matrix are presented in Figure 5 a and b, and Figure 6 respectively. Contrary to expectations, the alternative approach CNN-2 exhibited lower accuracy compared to the standard CNN, scoring at 96.5. Despite its faster execution time it boasts the largest memory footprint among all models. Furthermore, Figure 5 c and d showcase satisfactory accuracy, and loss curves.

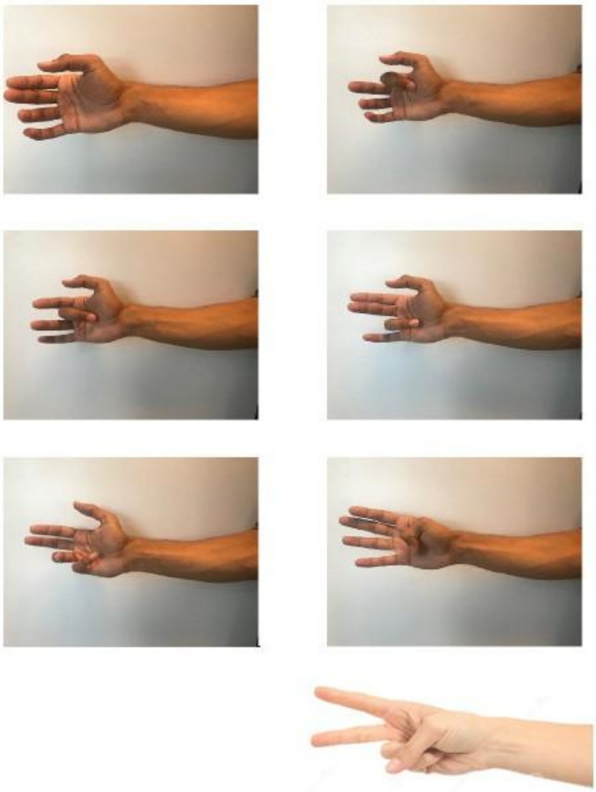
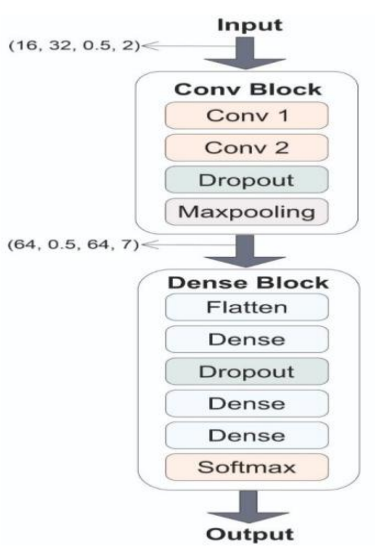
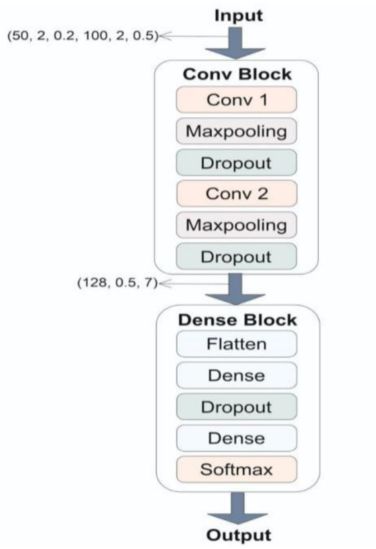


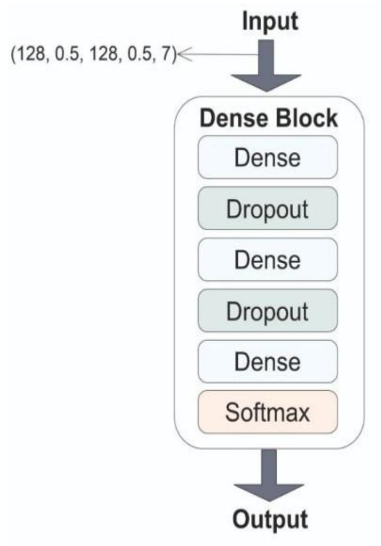
Fig. 2: (index finger, middle finger, ring finger, little finger, thumb, rest gestures) [17], victory gesture Source [29].



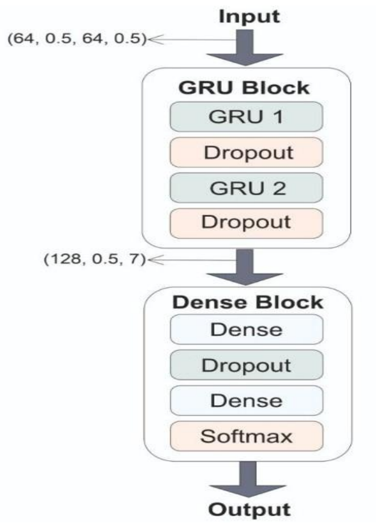
(a) CNN-1



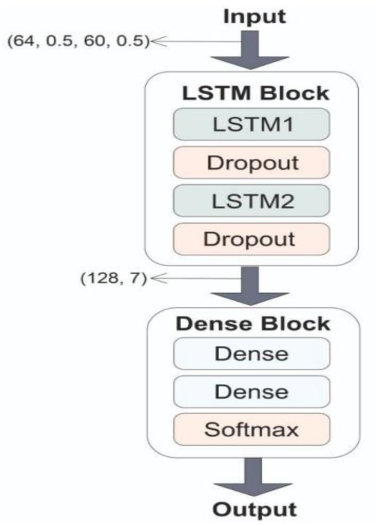
(b) CNN-2



(c) DFNN model



(d) GRU model



(e) LSTM model

Fig. 3: Models of CNN-1, CNN-2, DFNN, GRU, LSTM

Table 3: Hyperparameters of DL algorithms

hyperparameters	**CNN-1**	**CNN-2**	**DFNN**	**GRU**	**LSTM**
Epoch	100	28	100	100	60
Batch size	32	15	20	20	20
Optimizer	Adam	Adam	Adam	Adam	Adam
Activation function	Relu	Relu	Relu	Relu	Relu
Learning rate	0.001	0.001	0.001	0.001	0.001
Random state	42	0	42	42	42

Similarly, the proposed DFNN model attained slightly lower performance compared with the proposed CNN model with an accuracy of 96.2. Nevertheless, DFNN has better timing performance compared to CNN, Figures 5 e and f show the training accuracy and loss of the proposed DFNN. Moreover, the suggested GRU model out-outperformed the proposed DFNN and LSTM models with an accuracy of 96.8. On the other hand, Figure 5 g and h, respectively, show the training accuracy and loss of the proposed GRU. The suggested LSTM model outperformed the suggested DFNN model with an accuracy of 96.4. On the other hand, LSTM performs better in terms of timing than GRU Figure 5 i and j, show the training accuracy and loss of the proposed LSTM. According to Table 6, improved precision, recall, and F1-score are evident across all algorithms. Likewise, Table 7 summarizes the best accuracy results of the proposed work compared with related works.

4.3. Discussion

This section reports the outcomes of our comparative analysis of various neural network models, including CNNs, DFNN, and recurrent neural networks (GRU and LSTM).

The performance was assessed based on accuracy, computational time, memory usage, and optimizer parameters. Additionally, we discuss the hyperparameters set for each model to offer insights into the configuration that led to the observed performances. Contrasting to other studies, we investigated developing lightweight models for EMG classification. The models were configured with specific hyperparameters aimed at optimizing their performance. All models of DL used the Adam optimizer with varying learning rates and batch sizes. The consistency across most hyperparameters, such as the activation function (ReLU) and optimizer (Adam), allows for a more controlled comparison of architectural impacts on performance as shown in Table 3.

A lightweight model is of high interest, particularly in scenarios where resource constraints are a concern. This makes this method applicable in edge devices and reduces both energy and computational efficiency. In addition, a slight performance improvement is recorded when using combinations of feature extraction. The results indicate that employing appropriate hyperparameters and layers leads to better performance in classifying EMG signals. As indicated in Table 7, the random forest achieves a high classification accuracy of 95.9. SVM followed RF with 95.7 accuracy surpassing the performance of [17] and [30] which achieved 94.9. and 92.4 respectively.

The proposed CNN-1, requires 228,026 parameters with a memory requirement of only 890.73 KB, despite this huge saving, the accuracy is still intact with a value of 97.3. CNN-2 achieved a slightly lower accuracy of 96.6 compared to the standard CNN and has the advantage of having the lowest execution time at 0.66 minutes. However, this model has a higher memory usage at 2590.72 KB due to its significantly larger number of parameters 663,782. This trade-off suggests that the CNN-2 is efficient in terms of speed but requires more memory. This model is ideal for applications where time efficiency is paramount, and memory usage can be accommodated. Both CNN models achieved higher accuracy than [31] which achieved 90.8. In contrast, The DFNN model, with 80,666 parameters and 315.11 KB of memory, achieved 96.2 accuracies in 0.78 minutes in contrast to [32] and [17] which attained an accuracy of 95. Additionally, the GRU model, which requires 182,426 parameters and 712.61 KB of memory, achieved a slightly lower accuracy of 96.8 in 3.05 minutes compared to CNN-1. On the other hand, the LSTM model reached an accuracy of 96.4 with 221,738 parameters and 866.17 KB of memory in 1.92 minutes. The previous model required 60 epochs compared with [33] which achieved an accuracy of 99.6 in ten hours operated on CPU with 100 epochs.

In practical applications, lightweight neural network models are distinguished by their lower computational demands and reduced memory consumption. These attributes render them particularly suitable for integration into real-time systems, where efficiency and speed are paramount. Also, a lightweight model is crucial for EMG classification because it ensures real-time processing and efficient operation on devices with limited computational resources and battery life. This enhances portability and usability in applications such as prosthetics and wearable technology, i.e. a wearable bracelet has limited size and computation ability, and hence lightweight model is desired for EMG classification.

Table 4: Comparison of Machine Learning Model Performances Across Different Feature Sets

Features	XGB	Random Forest	SVM
SD	94.7	95.4	51.5
SD+RMS	95.1	94.3	76
SD +RMS +minimum	94.3	94.7	81.6
SD +RMS +minimum +maximum	95.1	94.4	85
SD +RMS +minimum +maximum + ZC	95.3	94.9	90.8
SD +RMS +minimum +maximum+ ZC + AAC	95.7	95.1	91.7
SD+ RMS +minimum +maximum + ZC + AAC + AFB	96	95.1	94.1
SD +RMS +minimum +maximum+ ZC + AAC + AFB + MAV	95.8	95.6	95
SD +RMS +minimum +maximum+ ZC + AAC + AFB + MAV + WL	96.2	95.9	95.1
SD +RMS +minimum +maximum+ ZC + AAC + AFB + MAV + WL + WAMP	95.7	94.8	95.7

Table 5: Comparative Analysis of Neural Network Models on Accuracy and Efficiency Metrics

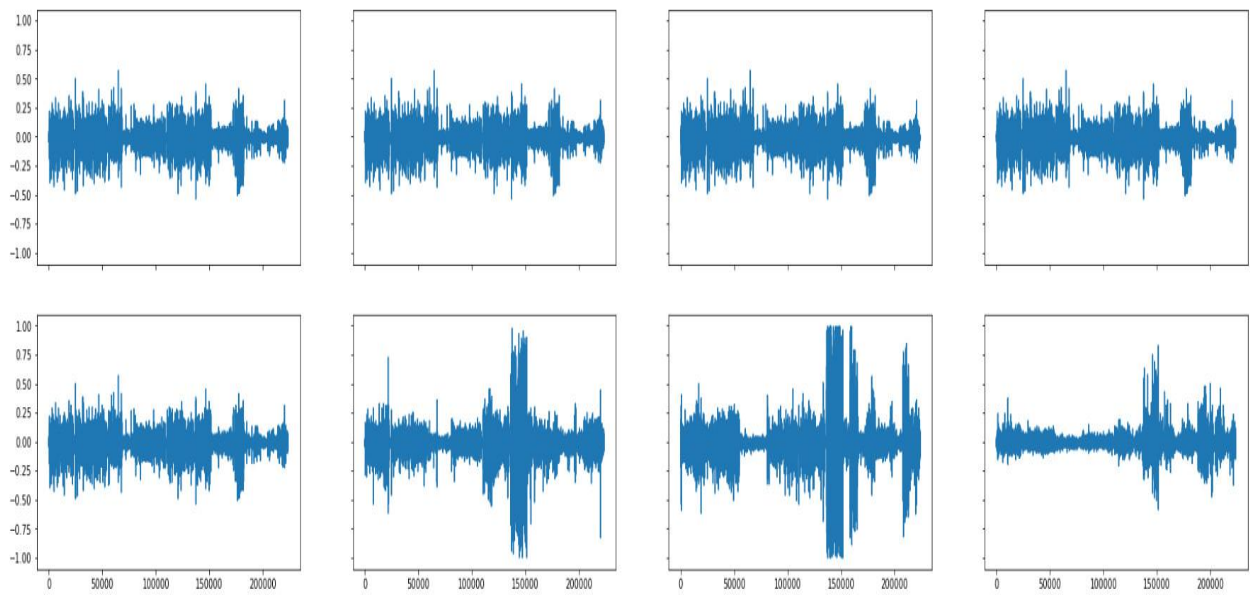
Algorithm	accuracy	time (m)	total parameters	memory size
CNN-1	97.3	0.96	228,026	890.73 KB
CNN-2	96.6	0.66	663,782	2590.72 KB
DFNN	96.2	0.78	80,666	315.11 KB
GRU	96.8	3.05	182,426	712.61 KB
LSTM	96.4	1.92	221,738	866.17 KB

Table 6: Performance Metrics Comparison Across Various Neural Networks

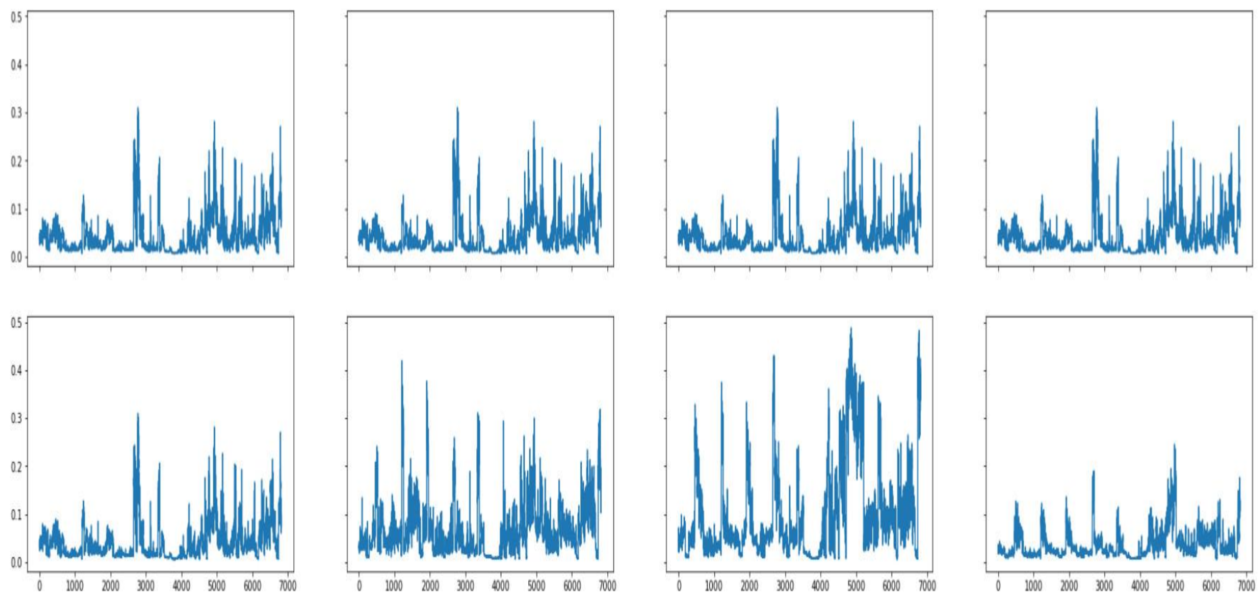
Algorithm	precision	recall	f1-score
CNN-1	97.3	97.3	97.3
CNN-2	97.3	97.3	97.3
DFNN	96.2	96.2	96.2
GRU	96.8	96.8	96.8
LSTM	96.5	96.4	96.4
XGB	96.2	96.2	96.2
RF	95.9	95.9	95.9
SVM	95.7	95.7	95.7

Table 7: Results of proposed work compared with related works

Algorithms	Proposed work accuracy	Related work
CNN-1	97.3	90.8 [31]
CNN-2	96.6	90.8 [31]
GRU	96.8	–
LSTM	96.4	99.6 [33]
DFNN	96.2	95 [17],[32]
XGB	96.2	–
RF	95.9	–
SVM	95.7	94,9 [17] 92.4 [30]

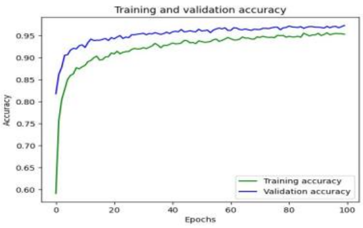


(a) Original Raw data

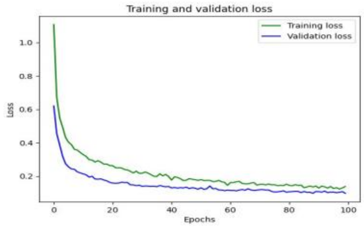


(b) Visualization of key features extracted from the raw data

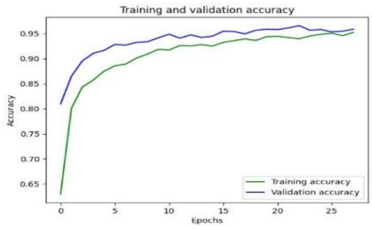
Fig. 4: Raw Data and Feature Extraction. The first subfigure represents the data from the first electrode, with subsequent subfigures representing the following electrodes. The x-axis represents samples per second while the y-axis indicates amplitude in millivolts.



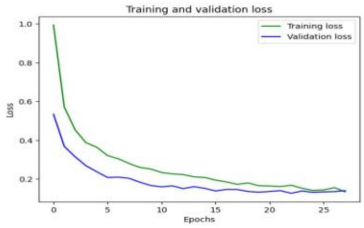
(a) Training and Validation accuracy of CNN-1



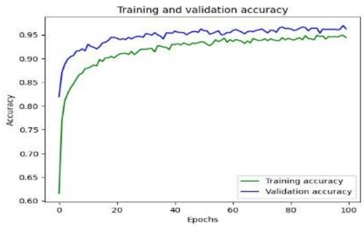
(b) Training and Validation loss of CNN-1



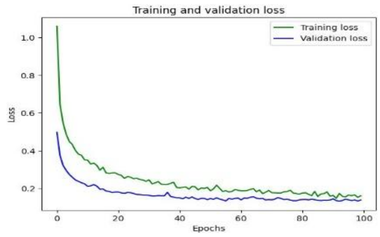
(c) Training and Validation accuracy of CNN-2



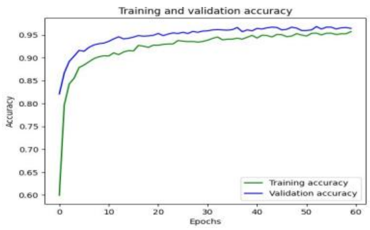
(d) Training and Validation loss of CNN-2



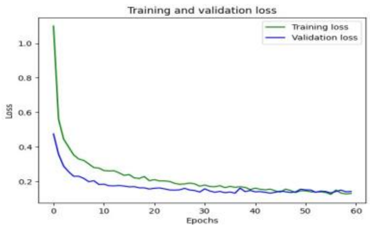
(e) Training and Validation accuracy of DFNN



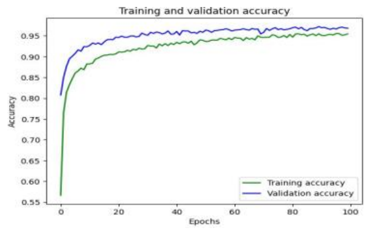
(f) Training and Validation loss of DFNN



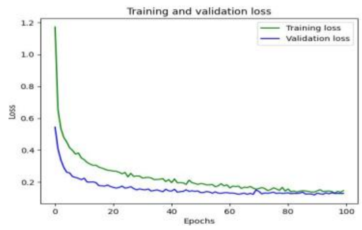
(g) Training and Validation accuracy of LSTM



(h) Training and Validation loss of LSTM



(i) Training and Validation accuracy of GRU



(j) Training and Validation loss of GRU

Fig. 5: Accuracy and loss of CNN-1, CNN-2, DFNN, LSTM and GRU

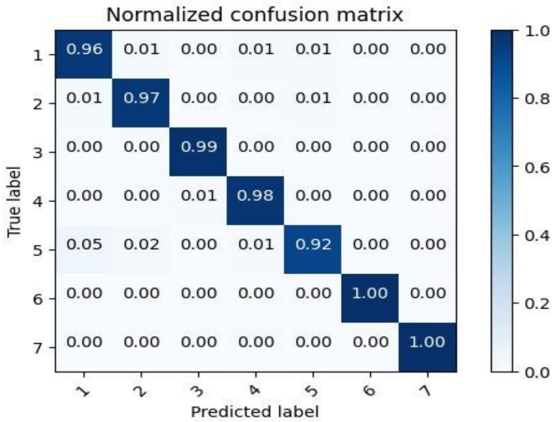


Fig.6: Best confusion matrix of CNN-1

5. CONCLUSIONS

This study has shown the practicality and efficacy of applying ML approaches for the classification of Electromyography (EMG) signals and significant differences were identified among classifiers. In this work, we evaluated the performance of various deep-learning algorithms. The CNN-1 model achieved the highest accuracy of 97.3% with a moderate computation time and memory usage, making it the most accurate model overall. CNN-2, while slightly less accurate 96.6%, had the fastest processing time but required the largest memory size. The DFNN, though slightly less accurate 96.2%, was the most efficient in terms of both computation time and memory usage. The GRU model demonstrated good accuracy 96.8% but had the longest computation time, while the LSTM model showed a balance between accuracy 96.4% and moderate computation time. Among the traditional machine learning algorithms, XGB performed best with an accuracy of 96.2%, indicating that deep learning models, particularly CNN-1, are superior in terms of classification performance for EMG signals, albeit with a trade-off in memory usage and computational efficiency. These results highlight the trade-offs between accuracy, computation time, parameter count, and memory usage in the selection of machine learning models for specific applications. Also, these results have broad applicability and can be effectively utilized in numerous electromyography (EMG) signal classification studies across various fields, including medical and engineering disciplines.

REFERENCES

[1] C. Tepe and M. Erdim, "Classification of emg finger data acquired with myo armband," in *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, 2020: IEEE, pp. 1-4.doi: 10.1109/HORA49412.2020.9152850

[2] A. Subasi, E. Yaman, Y. Somaily, H. A. Alynabawi, F. Alobaidi, and S. Altheibani, "Automated EMG signal classification for diagnosis of neuromuscular disorders using DWT and bagging," *Procedia Computer Science*, vol. 140, pp. 230-237, 2018.doi.org/10.1016/j.procs.2018.10.333

[3] G. Jia, H.-K. Lam, J. Liao, and R. Wang, "Classification of electromyographic hand gesture signals using machine learning techniques," *Neurocomputing*, vol. 401, pp. 236-248, 2020.doi.org/10.1016/j.neucom.2020.03.009

[4] Z. Taghizadeh, S. Rashidi, and A. Shalbaf, "Finger movements classification based on fractional fourier transform coefficients extracted from surface emg signals," *Biomedical Signal Processing and Control*, vol. 68, p. 102573, 2021.doi.org/10.1016/j.bspc.2021.102573

[5] E. Gokgoz and A. Subasi, "Comparison of decision tree algorithms for EMG signal classification using DWT," *Biomedical signal processing and control*, vol. 18, pp. 138-144, 2015.doi.org/10.1016/j.bspc.2014.12.005

[6] A. Phinyomark, P. Phukpattaranont, and C. Limsakul, "Feature reduction and selection for EMG signal classification," *Expert systems with applications*, vol. 39, no. 8, pp. 7420-7431, 2012.doi.org/10.1016/j.eswa.2012.01.102

[7] M. Jabbari, R. N. Khushaba, and K. Nazarpour, "EMG-based hand gesture classification with long short-term memory deep recurrent neural networks," in *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2020: IEEE, pp. 3302-3305. doi: 10.1109/EMBC44109.2020.9175279

[8] Z. Li, J. Zuo, Z. Han, X. Han, C. Sun, and Z. Wang, "Intelligent classification of multi-gesture EMG signals based on LSTM," in *2020 International Conference on Artificial Intelligence and Electromechanical Automation (AIEA)*, 2020: IEEE, pp. 62-65. doi: 10.1109/AIEA51086.2020.00020

[9] M. Aviles, J. M. Alvarez-Alvarado, J.-B. Robles-Ocampo, P. Y. Sevilla-Camacho, and J. Rodríguez-Reséndiz, "Optimizing RNNs for EMG Signal Classification: A Novel Strategy Using Grey Wolf Optimization," *Bioengineering*, vol. 11, no. 1, p. 77, 2024.doi.org/10.3390/bioengineering11010077

[10] C. K. Bhattachargee, N. Sikder, M. T. Hasan, and A.-A. Nahid, "Finger movement classification based on statistical and frequency features extracted from surface EMG signals," in *2019 International Conference on Computer, Communication, Chemical, Materials and Electronic Engineering (IC4ME2)*, 2019: IEEE, pp. 1-4. doi: 10.1109/IC4ME247184.2019.9036671

[11] A. Subasi and E. Yaman, "EMG signal classification using discrete wavelet transform and rotation forest," in *CMBEBIH 2019: Proceedings of the International Conference on Medical and Biological Engineering, 16–18 May 2019, Banja Luka, Bosnia and Herzegovina*, 2020: Springer, pp. 29-35.doi.org/10.1007/978-3-030-17971-7_5

[12] C. Millar, N. Siddique, and E. Kerr, "LSTM

classification of sEMG signals for individual finger movements using low cost wearable sensor," in *2020 International Symposium on Community-centric Systems (CcS)*, 2020: IEEE, pp. 1-8. doi: 10.1109/CcS49175.2020.9231515

[13] A. Subasi, "Classification of EMG signals using PSO optimized SVM for diagnosis of neuromuscular disorders," *Computers in biology and medicine,* vol. 43, no. 5, pp. 576-586, 2013.doi.org/10.1016/j.compbiomed.2013.01.020

[14] M. Simão, P. Neto, and O. Gibaru, "EMG-based online classification of gestures with recurrent neural networks," *Pattern Recognition Letters,* vol. 128, pp. 45-51, 2019.doi.org/10.1016/j.patrec.2019.07.021

[15] D. Huang and B. Chen, "Surface EMG decoding for hand gestures based on spectrogram and CNN-LSTM," in *2019 2nd China Symposium on Cognitive Computing and Hybrid Intelligence (CCHI)*, 2019: IEEE, pp. 123-126.doi: 10.1109/CCHI.2019.8901936

[16] V. B. Srinivasan, M. Islam, W. Zhang, and H. Ren, "Finger movement classification from myoelectricsignals using convolutional neural networks," in *2018 IEEE international conference on robotics and biomimetics (ROBIO)*, 2018: IEEE, pp. 1070-1075.doi: 10.1109/ROBIO.2018.8664807

[17] N. Naseer, F. Ali, S. Ahmed, S. Iftikhar, R. A. Khan, and H. Nazeer, "EMG based control of individual fingers of robotic hand," in *2018 International Conference on Sustainable Information Engineering and Technology (SIET)*, 2018: IEEE, pp. 6-9. doi: 10.1109/SIET.2018.8693177

[18] S. Krishnan, R. Akash, D. Kumar, R. Jain, K. M. M. Rathai, and S. Patil, "Finger movement pattern recognition from surface EMG signals using machine learning algorithms," in *ICTMI 2017: Proceedings of the International Conference on Translational Medicine and Imaging*, 2019: Springer, pp. 75-89. doi.org/10.1007/978-981-13-1477-3_7

[19] M. Fındık, Ş. Yılmaz, and M. Koseoglu, "Random forest classification of finger movements using electromyogram (emg) signals," in *2020 IEEE SENSORS*, 2020: IEEE, pp. 1-4. doi: 10.1109/SENSORS47125.2020.9278619

[20] M. V. Arteaga, J. C. Castiblanco, I. F. Mondragon, J. D. Colorado, and C. Alvarado-Rojas, "EMG-driven hand model based on the classification of individual finger movements," *Biomedical Signal Processing and Control,* vol. 58, p. 101834, 2020.doi.org/10.1016/j.bspc.2019.101834

[21] K. H. Lee, J. Y. Min, and S. Byun, "Electromyogram-based classification of hand and finger gestures usingartificial neural networks," *Sensors,* vol. 22, no. 1, p. 225, 2021.doi.org/10.3390/s22010225

[22] C. Tepe and M. Erdim, "Classification of surface electromyography and gyroscopic signals of finger gestures acquired by Myo armband using machine learning methods," *Biomedical Signal Processing and Control,* vol. 75, p. 103588, 2022.doi.org/10.1016/j.bspc.2022.103588

[23] G. Kumar, S. S. Yadav, and V. Pal, "Machine learning-based framework to predict finger movement for prosthetic hand," *IEEE Sensors Letters,* vol. 6, no. 6, pp. 1-4, 2022.doi: 10.1109/LSENS.2022.3147518

[24] A. Sultana, F. Ahmed, and M. S. Alam, "A systematic review on surface electromyography-based classification system for identifying hand and finger movements," *Healthcare Analytics,* vol. 3, p. 100126, 2023.doi.org/10.1016/j.health.2022.100126

[25] A. Sultana, M. T. I. Opu, F. Ahmed, and M. S. Alam, "A novel machine learning algorithm for finger movement classification from surface electromyogram signals using welch power estimation," *Healthcare Analytics,* vol. 5, p. 100296,*2024*.doi.org/10.1016/j.health.2023.100296

[26] N. M. Esa, A. M. Zain, and M. Bahari, "Electromyography (EMG) based classification of finger movements using SVM," *International Journal of Innovative Computing,* vol. 8, no. 3, 2018.doi.org/10.11113/ijic.v8n3.181

[27] J. Kobylarz, J. J. Bird, D. R. Faria, E. P. Ribeiro, and A. Ekárt, "Thumbs up, thumbs down: non-verbal human-robot interaction through real-time EMG classification via inductive and supervised transductive transfer learning," *Journal of Ambient Intelligence and Humanized Computing,* vol. 11, pp. 6021-6031, 2020. doi.org/10.1007/s12652-020-01852-z

[28] https://www.kaggle.com/datasets/nccvector/electromyography-emg-dataset

[29] https://www.dreamstime.com/royalty-free-stock-photos-woman-hand-scissors-gesture-image28257308

[30] R. Jain and V. K. 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, 2020.doi:10.18280/ria.340517

[31] H. O. Farag *et al.*, "Electromyography Signal Classification using Convolution Neural Network Architecture for Bionic Arm High Level Control," in *2021 16th International Conference on Computer Engineering and Systems (ICCES)*, 2021: IEEE, pp. 1-6.doi: 10.1109/ICCES54031.2021.9686171

[32] R. Jain and V. K. Garg, "EMG classification using nature-inspired computing and neural architecture," in *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)*, 2021: IEEE, pp. 1-5. doi: 10.1109/ICRITO51393.2021.9596077

[33] T. M. Bittibssi, M. A. Genedy, and S. A. Maged, "sEMG pattern recognition based on recurrent neural network," *Biomedical Signal Processing and Control,* vol. 70, p. 103048, 2021.doi: 10.1109/HORA49412.2020.9152850

تطوير تصنيف حركة إصبع EMG مع استخلاص الميزات والتعلم الآلي

رشا احمد ميسر
rasha.moysar2014@stu.uoninevah.edu.iq

محمد عبد المطلب عبدالله
mohammmed.abdulmuttaleb@uoninevah.edu.iq

هندسة الحاسوب والمعلوماتية، كلية هندسة الالكترونيات، جامعة نينوى، الموصل، العراق

تاريخ الاستلام: 5 أغسطس 2024 استلم بصيغته المنقحة: 7 أكتوبر 2024 تاريخ القبول:27 أكتوبر 2024

الخلاصة

في تطوير الأصابع الاصطناعية باستخدام بيانات مخطط كهربية العضل(EMG) ، يتمثل التحدي الحاسم في التعرف بدقة على حركات الأصابع، وبالتالي تتطلب نماذج مطورة تعالج إشاراتEMG ، وتسهل التصنيف المستقل لإيماءات الأصابع بدقة عالية. لتصنيف إشارة EMG بنجاح، يجب تقييم اختيار الميزة بعناية. ومع ذلك، فقد استخدمت العديد من الدراسات حول تصنيف إشارة EMG مجموعة ميزات تتضمن عدة عناصر زائدة عن الحاجة. في هذه الدراسة، يتم استخدام عدة مجموعات من ميزات المجال الزمني لتقليل إشارة .EMG بالإضافة إلى ذلك، تم اقتراح نموذجين من معماريات CNN (CNN-1،CNN-2), DFNN، LSTM، و GRUلتوفير دقة عالية مع الحد الأدنى من الحمل الحسابي والحد الأدنى من المعلمات. ومن خلال الاختيار الدقيق للنماذج وتحسين المعلمات الفائقة، تم تعزيز فعالية النماذج. تم تقييم النماذج بناءً على مقاييس الدقة والضبط والاستدعاء ودرجة .F1 ومن بين النماذج المقترحة، أسفرت CNN-1 عن توازن جيد من حيث الدقة والوقت الحسابي وحجم الذاكرة، حيث بلغت دقة 97.3 في 0.96 دقيقة مع حجم ذاكرة 890.73 كيلو بايت. علاوة على ذلك، أثبتت المقارنة مع أحدث الأعمال كفاءة الطريقة المقترحة.

الكلمات المفتاحية:

تخطيط كهربية العضل(EMG) ، التعلم الآلي(ML) ، الشبكة العصبية التلافيفية (CNN) ، الشبكة العصبية ذات التغذية العميقة (DFNN) ، الذاكرة الطويلة قصيرة المدى (LSTM)، الوحدات المتكررة المسورة (GRUs)