# Deep Recurrent Neural Network Approach with LSTM Structure for Hand Movement Recognition Using EMG Signals

Hajar Y. Alimam*
Electrical Engineering Department, Benha Faculty of Engineering, Benha, Egypt
hajar.alimam20@beng.bu.edu.eg

Wael A. Mohamed
Electrical Engineering Department, Benha Faculty of Engineering, Benha, Egypt
Wael.ahmed@bhit.bu.edu.eg

Ayman S. Selmy
Electrical Engineering Department, Benha Faculty of Engineering, Benha, Egypt
ayman.mohamed01@bhit.bu.edu.eg

## ABSTRACT

Due to the increasing number of amputees and the need to use prosthetics that simulate human limbs, an improved technique is proposed to classify hand gestures using Deep Recurrent Neural Networks (DRNN) based on the surface Electromyographic (sEMG) signal on the forearm. The implemented models are built on Feed-Forward Neural Networks (FFNN), Deep Recurrent Neural Networks (DRNN), and Long Short-Term Memory Networks (LSTM) using two types of datasets. They were recorded for four and seven motions, respectively. Both were written by MYO armband, and the conception of the technique is divided into two main phases applied to the two types of datasets. Two DRNN models are implemented, the First is a multi-classifications DRNN with all dataset files imported simultaneously. Each data file is then imported separately as input to the second binary classification DRNN model. Classification results for the multi-DRNN classifier and binary one is compared according to both datasets separately. Results show that the average accuracy for multi-classifications was (95%, and 86%) for both datasets while binary classification was 99% accurate for each model. Additionally, precision, recall, and f1-score were determined for both datasets, yielding better results.

## KEYWORDS

Keywords. Prosthetics, Hand gesture classification, Deep Recurrent Neural Networks (DRNN), Surface Electromyographic signal (sEMG), Long Short-Term Memory networks (LSTM), Prosthetic limbs

**ACM Reference Format:**
Hajar Y. Alimam, Wael A. Mohamed, and Ayman S. Selmy. 2023. Deep Recurrent Neural Network Approach with LSTM Structure for Hand Movement Recognition Using EMG Signals. In *2023 12th International Conference on Software and Information Engineering (ICSIE) (ICSIE 2023), November 21–23, 2023, Sharm El-Sheikh, Egypt*. ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3634848.3634851

*Corresponding author.


*ICSIE 2023, November 21–23, 2023, Sharm El-Sheikh, Egypt*

ACM ISBN 979-8-4007-0810-7/23/11…$15.00
https://doi.org/10.1145/3634848.3634851

## 1 INTRODUCTION

Amputees are people who have lost their upper limbs due to accidents, trauma, or diseases that affect the limbs. Such a person cannot perform functions that require the use of the arm [1]. The desire of amputees to preserve their lost arm with an artificial device is a major driving force in research into the improvement of prosthetic technology. Myoelectric prostheses. It is a human-machine interface controlled using (sEMG) signals collected from the remaining muscle tissues on the residual limb of an amputee [2, 3]. It is used to communicate and control Human-Machine Interactions (HMI) by recognizing specific patterns of activity and translating those patterns into meaningful control commands [4].

sEMG is the study of electrical muscle signals recorded on the skin surface. sEMG is used for clinical diagnosis, muscle fatigue analysis, rehabilitation, and prosthesis control. sEMG is a more popular method of HMI measurements as it can be performed directly and is non-invasive. performed by a trained person, not necessarily a physician, with minimal risk [5].

Improvement of the pattern recognition system is necessary to control myoelectric prostheses with high accuracy. Many attempts have been made in this regard using various methods of hand gesture classification based on Artificial Intelligence (AI) [6].

AI is the technique of representing human intellectual skills in machines, including Machine Learning (ML) and Deep Learning (DL) [7]. ML and DL algorithms applied to EMG signals are an expanding and rapidly growing research field, allowing researchers to learn HMI to be experts and later help decision-making processes [8]. The most commonly used models include Support Vector Machines (SVM), Random Forest, K-Nearest Neighbors (KNN), Artificial Neural Networks (ANN), Convolution Neural Networks (CNN), and Recurrent Neural Networks (RNN) [9, 10].

Many deep neural network algorithms for gesture recognition have been presented by researchers during the past few years. Nguyen et al. [11] presented an implementation of a RNN deployed on NVIDIA Jetson Nano. This makes it possible to implement neuro prostheses as wearable, self-contained units that can control finger movements individually in real-time. Preliminary results show accuracy (95% to 99%). Aly et al. [12] suggested Three deep learning models for hybrid signal classification systems, including the CNN model, the LSTM model, and the combined CNN-LSTM model. To evaluate the experiment, a data set of multiple-channel electroencephalography (EEG) signals was merged with multiple-channel sEMG signals that decode hand and wrist motions. Preliminary test results show accuracy (91% to 93.5%). Dolopikos et al. [13] Random Forest, SVM, a Multi-layer Perceptron, and Deep Neural Network (DNN) are some of the classifiers implemented in this

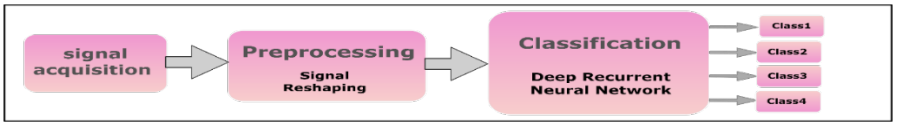


**Figure 1: Proposed hand gestures classification technique block diagram.**

study. An ensemble voting mechanism is used to integrate the three classical classifiers into a single model, scoring 91.93%, compared to the DNN's performance of 88.68%. Jafarzadeh et al. [14] provide a DL method for manipulating prosthetic hands using their raw EMG signals. By using a novel deep CNN, feature engineering steps can be avoided. A step towards an end-to-end optimization paradigm is feature extraction and feature description elimination. The accuracy achieved is 91.26%. A. Gautam et al. [15] described the Low-Complex Movement Recognition-Net (LoCoMo-Net) DL framework in order to recognize wrist and finger flexion. Force patterns from single-channel (sEMG) recordings of grasping and functioning motions. The average classification accuracies achieved by LoCoMo-Net model for all movements in 11 healthy subjects, 3 amputees, were 93.4%, 88.8%, and 94.7%, respectively. In Abu et al. [16] presented EMG signal classification of multiple hand gestures based on ANN. Each movement's EMG pattern was extracted through signal feature extraction and used to train a classifier. ANN achieved 80% accuracy in training and testing for 10 hidden layers. Oh et al. [17] For classification hand gestures, apply a simple CNN model with raw EMG, Short-Time Fourier Transform (STFT), Wavelet Transform (WT), and Scale-Averaged Wavelet Transform (SAWT), and compare the performance of Using CNN with SAWT can improve the accuracy of selected hand gestures by up to 94.6%, which is more accurate and less computationally efficient than traditional multi-channel STFT or WT. (Nahid et al. [18] introduces classifying hand movement from two known sEMG datasets. Applying Transfer Learning (TL) and CNN-LSTM. A combination of CNN and LSTM has achieved accuracy for these data. 99.72% for the first data type and 99.83% for the second, respectively.

This study's main contribution is the development of a new EMG-based control system for arm prostheses. The proposed method is based on FFNN, DRNN, and LSTM and uses two types of datasets, without performing any feature extraction, or spectrogram analysis.

Figure 1 represents the proposed method workflow as; initially sEMG signal was recorded by using MYO armband for different hand gestures, then we applied reshaping for data as a simple signal processing, then we used our proposed technique for hand gestures classification. Finally, the system is ready for controlling the limb.

The rest of the article is organized as follows: Initially, section 2 presents the proposed classification approach and datasets used. Then section 3 presents the results of the proposed method. The final section is a conclusion to the work.

## 2 MATERIALS AND METHODS

### 2.1 Data acquisition

In this study, we investigate the possibility of detecting two different types of arm movements using DRNN. This paper uses two open access datasets. Both were recorded with MYO arm band which has eight sensors placed on the surface of the skin. Each dataset row contains 8 consecutive readings from all 8 sensors. That is, 64 columns of EMG data. The last column is the result of the gestures made during data recording. Initial data has four gestures recorded (ok, paper, scissors, and rocks). The second was recorded with seven gestures (resting hand, clenched hand, wrist flexion, wrist extension, radial deviation, ulnar deviation, and palm extension).

The first data was recorded at 200 Hz. So, each line is 8*(1/200) seconds = 40 milliseconds of recording time. A classifier with 64 digits predicts gesture classes (0 to 3). Each record for a certain gesture class is combined into a CSV file with an appropriate name (0-3). Here is the gesture class: Rock-0, Scissors-1, Paper-2, OK-3. [19].

For the second data, raw EMG data are provided from 36 subjects while they performed a series of static hand movements. Each gesture was performed for 3 seconds with a 3-second pause between gestures. Gesture classes were Resting hand-0, Clenched hand-1, Wrist flexion-2, Wrist extension-3, Radial deviation-4, Lunar deviation-5, Palm extension-6. Each record for a certain gesture class is combined into a CSV file with an appropriate name (0-6) [20].

### 2.2 Methods

*2.2.1 Phase 1.* All datasets files are imported to the multi-classification DRNN model, and the accuracy is checked for two data types as shown in Figure 2. Figure 2 explains that all datasets file imported to the DRNN model as input for predicting multi-classes.

*2.2.2 Phase 2.* Each data file is imported as input to the binary classification DRNN module for predicting one class, and the accuracy is checked for two data types. Figure 3 shows that the rock gesture file was imported to Binary Classification Module (BCM) for predicting if the gesture is rock or not. And repeated for rest gestures. Then all BCM are fused together to ensure that the correct class is recognized as shown in Figure 4.

*2.2.3 Deep learning.* In order to transform data in meaningful ways and learn useful representations of input data, deep learning for neural networks assembles networks with multiple processing layers [21]. Sequential layers of representation are the core concept behind DL. This reduces the need for feature engineering for flat networks by enabling hierarchically learning of high-level features from low-level features. Because of this, DL is frequently referred to as a "hierarchical learning representation" in literature in some contexts [22, 23]. For classification, the model was implemented based on Deep Recurrent Neural Network (DRNN).

*2.2.4 Deep Recurrent Neural Networks.* DRNN is an FFNN extension that can accept inputs of variable-length sequences. The fact that DRNN contains a recurrent hidden state whose activation at

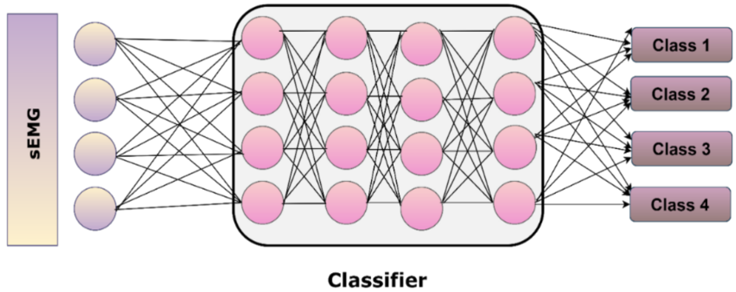


**Figure 2: Multi-classification DRNN model diagram.**

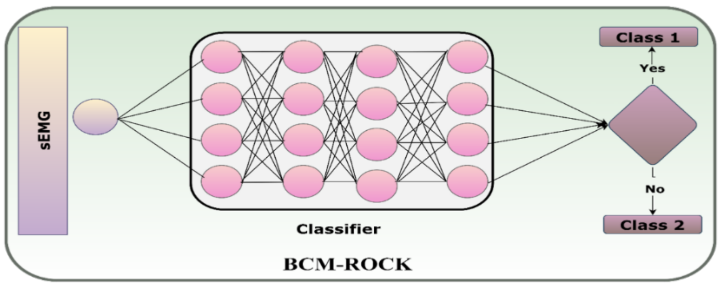


**Figure 3: Binary classification DRNN module diagram.**

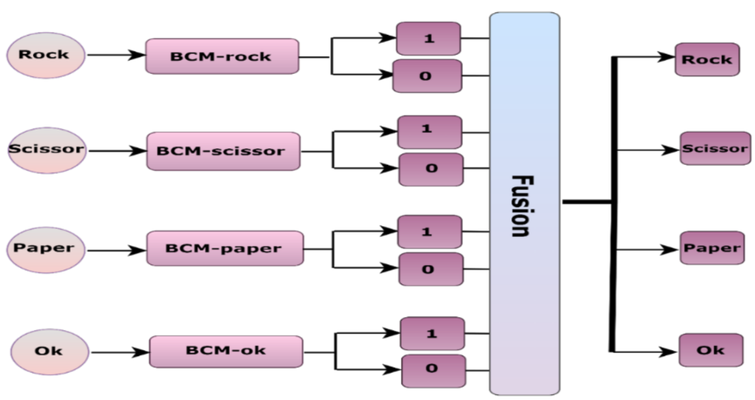


**Figure 4: Binary classification fusion diagram.**

each time depends on that, the previous time is the reason it can handle time series. One DRNN type that enables each recurrent unit to adaptively capture dependencies on various time scales is the LSTM [23]. To regulate the flow of information, LSTMs use cells and forget gates. This allows the model to accept a set of data samples as input and determine the time relationship between them. It has been observed, however, that they struggle to learn long-term partnerships. LSTM networks address this issue by including a parameter in the hidden node loop that allows states to be recorded and released based on the input sequence. Thus, states are triggered in response to short-term events, but the network can maintain these states active indefinitely, providing long-term memory to the network. In learning sequences, LSTMs have been proven to outperform regular DRNNs [24, 25].

Figure 5 represents the workflow of the proposed DRNN model and shows the number of LSTM and Dense layers used. Figure 6 shows the structure of LSTM cell and how cells are connected recurrently to each other.

*2.2.5 Stacked LSTM layers.* (LSTMs) units that learn long-term dependencies between time steps in temporal and sequence data make up a recurrent layer [23]. A reliable method for solving difficult sequence prediction problems is stacking LSTMs. A model of LSTM made up of several LSTMs layers is referred to as a "stacked" LSTM architecture. Instead of a single value output, the LSTMs

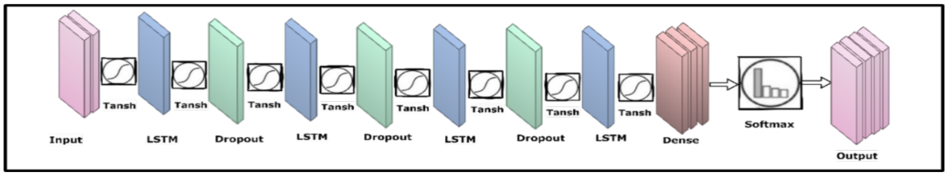


**Figure 5: Proposed DRNN layering architecture.**

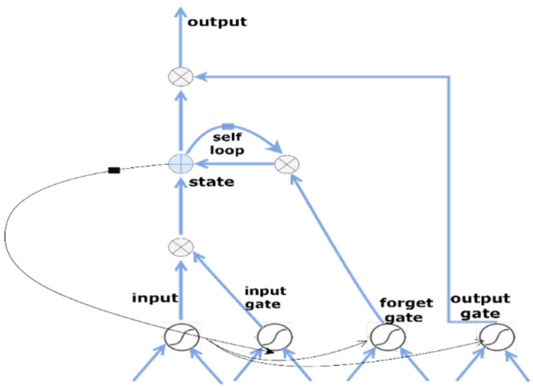


**Figure 6: Structure of LSTM unit**

layer below receives a sequence output from the layer above. To be more precise, one output time step for each input time step, as opposed to the opposite [24-27].

The LSTM network retrieves the mapping from the input sequence x = ($x_1, \ldots\ldots x_t$) to the output y = ($y_1 \ldots\ldots y_t$) by iteratively calculating the net unit activation from 1 to t as follows:

$$i_t = \sigma\left(W_i\, x_i + U_I h_{t-1} + b_i\right) \tag{1}$$

$$z_t = tanh\left(W_z x_t + U_z h_{t\,-1} + b_z\right) \tag{2}$$

$$f_t = \sigma\left(W_f x_t + U_f\, h_{t\,-1} + b_f\right) \tag{3}$$

$$C_t = \left(i_t * z_t\right) + \left(f_t * C_{t\,-1}\right) \tag{4}$$

$$o_t = \sigma\left(W_o x_t + U_o h_{t\,-1} + V_o h_t + b_o\right) \tag{5}$$

$$h_t = o_t * \tan h\left(C_t\right) \tag{6}$$

$$x_t = \tan h\left(W_i\, x_i + b_i\right) \tag{7}$$

Where: $f_t$ indicates the forget gate, $i_t$ indicates the input gate, $o_t$ indicates the output gate, $C_t$ indicates the cell state, $h_t$ indicates the hidden state, W represents each LSTM gate's unit weight, and b represents each LSTM gate's unit bias.

### 2.2.6 *Dense layers.*

Dense layers are fully connected layers that are used in the final stages of neural networks. These layers help change the dimensions of the output from the previous layer. This allows the model to easily define relationships between values in the data on which it operates. This is the most common and most used layer. The equation shows the vectorized representation of the dense layer [23, 24].

$$x_t = \tan h\left(W_i\, x_i + b_i\right) \tag{8}$$

Where: W represents the LSTM unit weights for each gate and b represents the LSTM unit bias for each gate.

### 2.2.7 *SOFTMAX activation function.*

SOFTMAX activation function are integrated into the output layer to classify which gestures are performed. One of the most commonly used functions is the SOFTMAX activation function. because of the performance of each class in predicting probability [23, 24]. The SOFTMAX function predicts a class's chance. The equation shows its mathematical expression.

$$Z_j = \frac{e^{Z_j}}{\sum_{j=1}^{K} e^{Z_j}} \text{ for } j = 1, \ldots.., K \tag{9}$$

$Z_j$ is the class's output probability, $K$ is the total number of predicted classes, $e^{Z_j}$ is the linear combination of the weights and previous layer activations. It is important to maintain that the SOFTMAX output layer has the same number of units or nodes as the predicted classes.

### 2.2.8 *Learning.*

The essential step after defining the model was selecting a cost function for optimizing during training and attaining the desired results [23, 24]. The Cost function used in the classification problem was the cross-entropy loss function which is defined with the given equation [28].

$$C = -\sum_{k=1}^{k} y_k\, log\left(P_k\right) \tag{10}$$

Where: $P_k$is neural network prediction, $y_k$indicates the actual value, $k$ indicates the total number of classes.

The model was optimized with ADAM optimization. ADAM is one of the best deep learning optimization methods, and its popularity is rapidly expanding [23, 24]. The ADAM optimizer has several advantages, including faster run-time, less memory requirements, and less fine-tuning than any other optimization algorithm [29]. One of ADAM's key elements is the use of an exponentially weighted moving average (leaky averaging) to obtain estimates of both the momentum and the second moment of the gradient. i.e., use a state variable.

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1)\left[\frac{\delta L}{\delta w_t}\right] \tag{11}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2)\left[\frac{\delta L}{\delta w_t}\right]^2 \tag{12}$$

Where: $\beta_1$ and $\beta_2$ are hyper-parameters which represent the decay rate of the average of the gradients.

### 2.2.9 *performance metrics.*

As the model has been created and trained, its performance must be evaluated. There are many metrics available to measure a model's performance. i.e., accuracy, precision, recall, f1-score [23, 30].

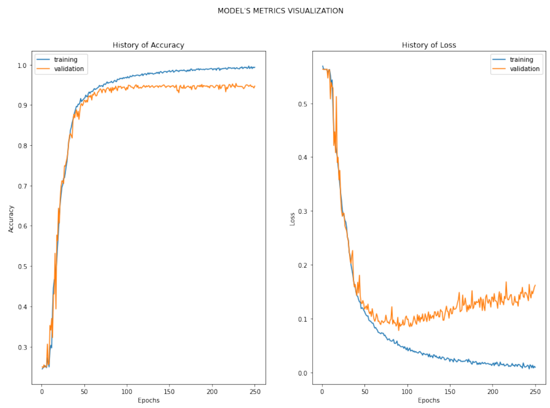


**Figure 7: Training and validation accuracy for data 1**

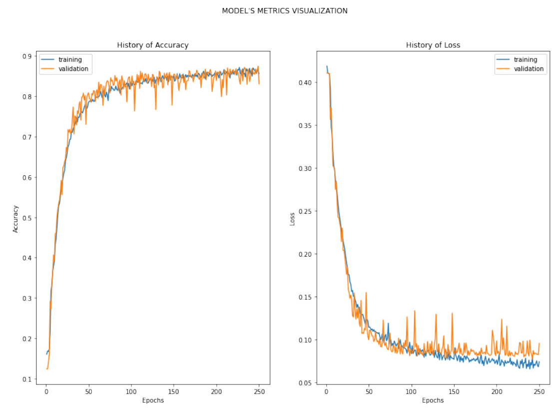


**Figure 9: Training and validation accuracy for data 2**

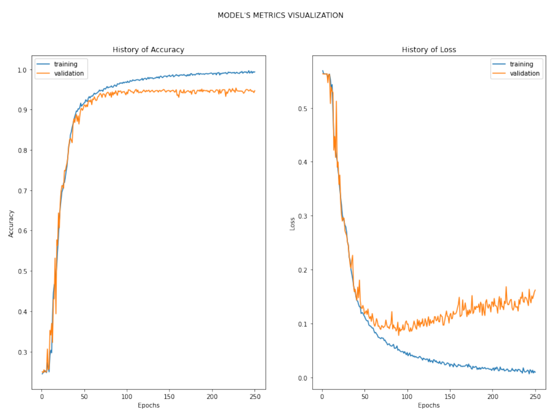


**Figure 8: Training and validation loss for data 1**

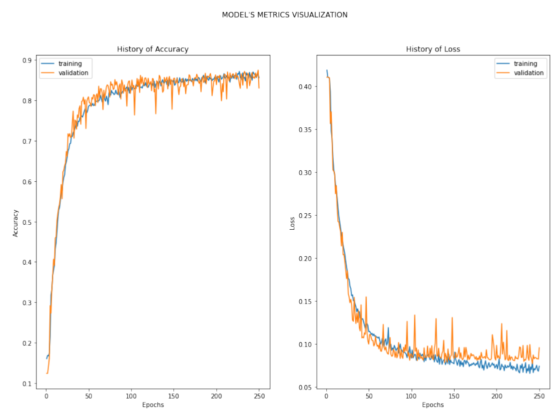


**Figure 10: Training and validation loss for data 2**

## 3 RESULTS AND DISCUSSIONS

Based on the methodology described in Section 2, this section provides a summary of the research results.

### 3.1 Multi-classification evaluation

For better performance and stability during training and validation, the number of training epochs was set to 250 and the batch size to 32. The correct classification rates (CCR) of the proposed method for first and second data are 99% and 86%, respectively. Figures 7,8,9 and 10 show the accuracy and the loss results during the training and the validation stages. Table 1 shows the classifier accuracy and loss values of training and testing. Figure 10,11 show the values of precision, recall and f1-score for both datasets. It can be noted that the performance of the multi classification model using the first data type is better than the second one.

### 3.2 Binary classification evaluation

The same for binary classification, the number of training epochs was set to 250 and the batch size to 32. The (CCR) of the proposed method is 99.55%. Figure 13 shows the values of precision, recall, and f1-score applied for binary classification.

**Table 1: accuracy and loss values of test and training for data 1&2**

| Data | Test Accuracy | Test Loss | Training Accuracy | Training Loss |
|---|---|---|---|---|
| Data1 | 0.946 | 0.16 | 0.99 | 0.01 |
| Data2 | 0.86 | 0.08 | 0.86 | 0.07 |

**Table 2: accuracy and loss values of test and training for binary classification**

| Test Accuracy | Test Loss | Training Accuracy | Training Loss |
|---|---|---|---|
| 0.946 | 0.16 | 0.99 | 0.01 |

Figure 14,15 shows the accuracy and loss results during the training and validation stages. Table 2 shows the classifier accuracy and loss value of training and testing. The results for each BCM achieve almost the same values.

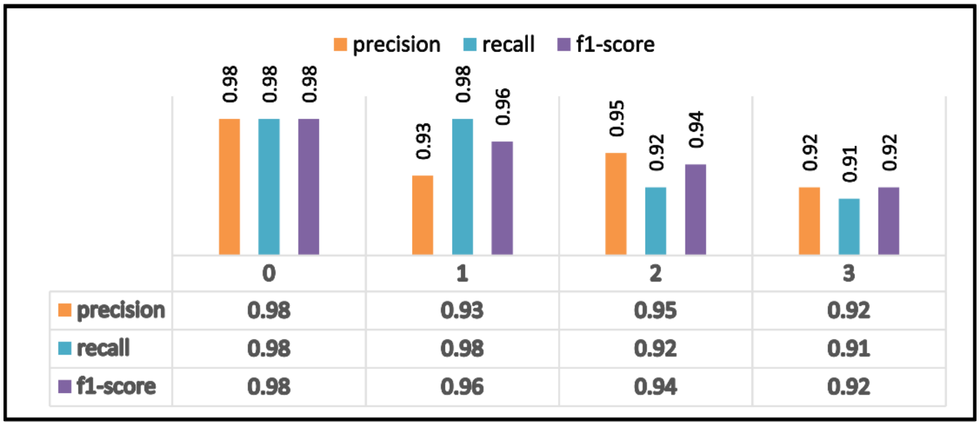


**Figure 11: Performance metrics results for multi-classification for data 1**

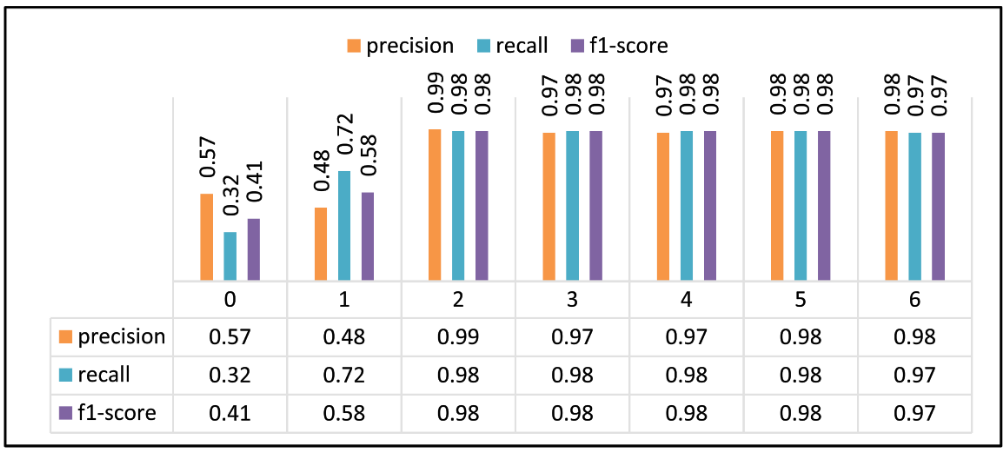


**Figure 12: Performance metrics results for multi-classification for data 1**

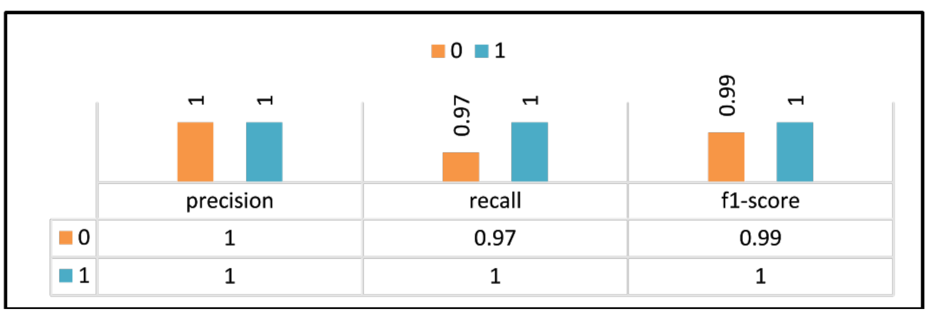


**Figure 13: Performance metrics results for binary classification**

## 3.3 Model comparison

For evaluating the performance of the proposed technique, the results of the two phases of the method stages were compared with each other and with previous works of the same purpose.

Figure 16 shows the comparison resultant between the accuracy of multi and binary classification for two data types. Table 3 shows the results of the evaluation of the proposed DRNN model compared with others using the same data. Looking at the results shown, it can be noticed that the performance of the binary classification is the best.

**Table 3: Evaluation of the proposed DRNN model with other models**

| Classifier | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|
| Catboost | .96 | .95 | .97 | .96 |
| Support vector | .92 | .94 | .90 | .92 |
| Random forest | .92 | .95 | .91 | .93 |
| Proposed technique (multi-classification) | .95 | .95 | .97 | .99 |
| Proposed technique (Binary-classification) | .99 | 1 | 1 | .99 |

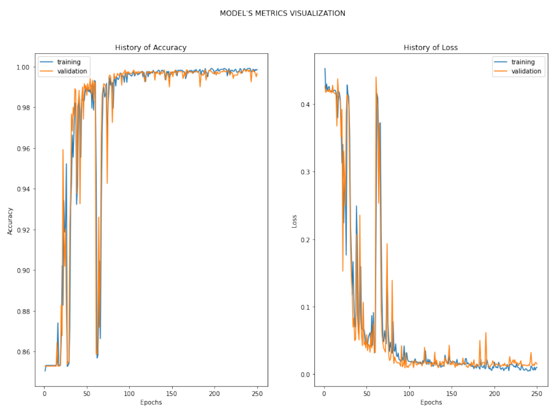


**Figure 14: Training and validation accuracy for data 1**

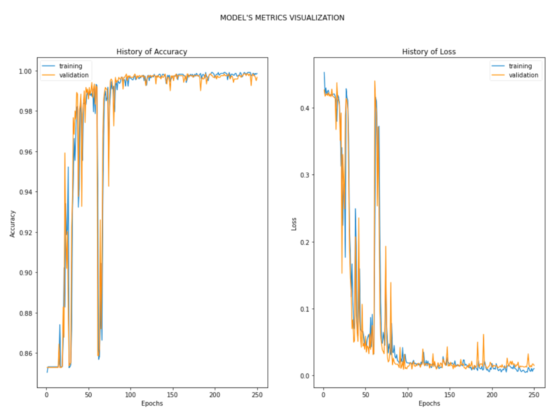


**Figure 15: Training and validation loss for data 1**

## 4 CONCLUSION

The usage of Deep Recurrent Neural Networks (DRNN) based on FeedForward Neural Networks (FFNN) and long short-term memory networks (LSTM) was proposed for this study to enhance the classification of hand gestures using EMG signals recorded for the forearm muscles. The main concept of the technique is divided into two phases multi and binary classification applied to the two types of datasets. Results show that the average accuracy for multiple classifications was 95% for the first data type and 86% for the second one, while binary classification was 99% accurate for each model. The accuracy of gestures classification performed better in the binary classification phase than in the multiple. The results obtained in two phases for two data types were compared with other models using the same data.

## REFERENCES

[1] Roșca AC, Baciu CC, Burtăverde V, Mateizer A. Psychological consequences in patients with amputation of a limb. An Interpretative-Phenomenological Analysis. Frontiers in Psychology. 2021;12.
[2] Grushko S, Spurný T, Cerny M. Control Methods for Transradial Prostheses Based on Remnant Muscle Activity and Its Relationship with Proprioceptive Feedback. Sensors. 2020;20(17):4883.
[3] Marinelli A, Boccardo N, Tessari F, *et al.* Active upper limb prostheses: a review on current state and upcoming breakthroughs. Progress in Biomedical Engineering. 2023;5(1):012001.
[4] Esposito D, Centracchio J, Andreozzi E, Gargiulo G, Naik GR, Bifulco P. Biosignal-Based Human–Machine Interfaces for Assistance and Rehabilitation: a survey. Sensors. 2021;21(20):6863.
[5] Del Olmo M, Domingo R. EMG Characterization and Processing in Production engineering. Materials. 2020;13(24):5815. doi:10.3390/ma13245815.
[6] Campbell E, Phinyomark A, Scheme E. Current trends and confounding factors in myoelectric control: limb position and contraction intensity. Sensors. 2020;20(6):1613.
[7] Nayak S, Das RK. Application of artificial intelligence (AI) in prosthetic and orthotic rehabilitation. In: IntechOpen EBooks. ; 2020.
[8] Zhang S, Suresh LP, Yang J, Zhang X, Tan SC. Augmenting Sensor Performance with Machine Learning Towards Smart Wearable Sensing Electronic Systems. Advanced Intelligent Systems. 2022;4(4):2100194
[9] Gopal P, Gesta A, Mohebbi A. A systematic study on Electromyography-Based hand gesture recognition for assistive robots using deep learning and machine learning models. Sensors. 2022;22(10):3650.
[10] Xiong D, Zhang D, Zhao X, Zhao Y. Deep Learning for EMG-based Human-Machine Interaction: A review. IEEE/CAA Journal of Automatica Sinica. 2021;8(3):512-533.
[11] Nguyen AT, Drealan MW, Luu DK, *et al.* A portable, self-contained neuroprosthetic hand with deep learning-based finger control. Journal of Neural Engineering. 2021;18(5):056051.
[12] Aly H, Youssef SM. Bio-signal based motion control system using deep learning models: a deep learning approach for motion classification using EEG and EMG signal fusion. Journal of Ambient Intelligence and Humanized Computing. 2021;14(2):991-1002.
[13] Dolopikos C, Pritchard M, Bird JJ, Faria DR. Electromyography Signal-Based gesture recognition for Human-Machine interaction in Real-Time through model calibration. In: Springer EBooks. ; 2021:898-914.
[14] Jafarzadeh M, Hussey DC, Tadesse Y. Deep learning approach to control of prosthetic hands with electromyography signals. IEEE International Symposium on Measurement and Control in Robotics (ISMCR. Published online September 1, 2019.
[15] Gautam A, Panwar M, Wankhede A, *et al.* Locomo-Net: a low -Complex deep learning framework for SEMG-Based hand movement recognition for prosthetic control. IEEE Journal of Translational Engineering in Health and Medicine. 2020;8:2100812.
[16] Abu MS, Rosleesham S, Suboh MZ, Yid MSM, Kornain Z, Jamaluddin NF. Classification of EMG signal for multiple hand gestures based on neural network. Indonesian Journal of Electrical Engineering and Computer Science. 2020;17(1):256.
[17] Oh DC, Jo YU. Classification of hand gestures based on multi-channel EMG by scale average wavelet transform and convolutional neural network. International Journal of Control Automation and Systems. 2021;19(3):1443-1450.
[18] Nahid N, Rahman A, Ahad MdAR. Deep Learning Based Surface EMG Hand Gesture Classification for Low-Cost Myoelectric Prosthetic Hand. 2020 Joint 9th International Conference on Informatics, Electronics & Vision (ICIEV) and 2020 4th International Conference on Imaging, Vision & Pattern Recognition (icIVPR),. Published online August 26, 2020.
[19] Yashuk K, Perikov I, classify gestures by reading muscle activity. (n.d.). Version 2 2018. | Kaggle. https://www.kaggle.com/datasets/kyr7plus/emg-4.

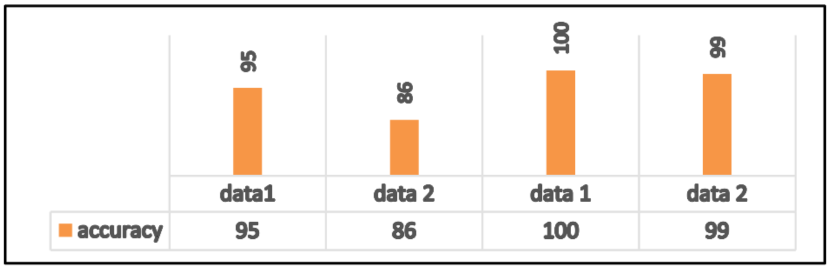


**Figure 16: Classification accuracy comparison**


[20] Frank A. UCI machine learning repository. EMG data for gestures DataSet.(n.d.).https://archive.ics.uci.edu/ml/datasets/EMG+data+for+gestures.

[21] Baduge SK, Thilakarathna S, Perera JM, *et al.* Artificial intelligence and smart vision for building and construction 4.0: Machine and deep learning methods and applications. Automation in Construction. 2022;141:104440.

[22] Moradi R, Berangi R, Minaei B. A survey of regularization strategies for deep models. Artificial Intelligence Review. 2019;53(6):3947-3986.

[23] Heaton J. Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning. Genetic Programming and Evolvable Machines. 2017;19(1-2):305-307.

[24] Toro-Ossaba A, Jaramillo-Tigreros J, Tejada JC, Pena A, González AL, Castanho RA. LSTM Recurrent Neural Network for hand gesture recognition using EMG signals. Applied Sciences. 2022;12(19):9700.

[25] Nadjib BL, Bilal C, Rebai K. EMG-Based Hand gesture recognition for myoelectric prosthetic hand control. 2021 International Conference on Artificial Intelligence for Cyber Security Systems and Privacy (AI-CSP). Published online November 20, 2021.

[26] Koch P, Dreier M, Maass M, Phan H, Mertins A. RNN With Stacked Architecture for sEMG based Sequence-to-Sequence Hand Gesture Recognition. ," 2020 28th European Signal Processing Conference (EUSIPCO), Amsterdam, Netherlands, 2021. Published online January 24, 2021.

[27] Ameur S, Khalifa AB, Bouhlel MS. A novel hybrid bidirectional unidirectional LSTM network for dynamic hand gesture recognition with Leap Motion. Entertainment Computing. 2020;35:100373.

[28] Li W, Shi P, Yu H. Gesture recognition using surface electromyography and deep learning for prostheses Hand: State-of-the-Art, Challenges, and Future. Frontiers in Neuroscience. 2021;15.

[29] Arora S, Gupta A, Jain R, Nayyar A. Optimization of the CNN model for hand sign language recognition using ADAM Optimization Technique. In: Lecture Notes in Networks and Systems. ; 2021:89-104.

[30] Fricke C, Alizadeh J, Zakhary N, Woost TB, Bogdan M, Classen J. Evaluation of three machine learning algorithms for the automatic classification of EMG patterns in GAIT disorders. Frontiers in Neurology. 2021;12.