Performance Evaluation of Temporal and Frequential Analysis Approaches of Electromyographic Signals for Gestures Recognition using Neural Networks

Edwar Jacinto Gómez, Fredy H. Martínez Sarmiento, Fernando Martínez Santa
Technology Faculty, Universidad Distrital Francisco José de Caldas, Bogotá, Colombia

***Abstract*—Now-a-days, human-machine interfaces are increasingly intuitive and straightforward to design, but there is difficulty capturing electromyographic signal data using the least amount of hardware. This work takes the signals of a human forearm as input parameters describing a series of five gestures, using a dataset of 8 channels of electromyographic signals, using as a capture device a Thalmic Labs Inc. handle called Myo armband. The aim is to compare the performance of the artificial neural network using data in the time domain as input to the learning system. The same data are pre-processed to the frequency domain, looking for an improvement in the neural network's performance since transforming the input signals of the system to the frequency domain minimizes the problems inherent to this type of signal. This transformation is achieved using the fast Fourier transform. Consequently, it seeks to reach a neural network architecture that recognizes the gestures captured with the Myo armband in a high percentage of performance to be used in stand-alone applications, using the TensorFlow libraries of Python for its design. As a result, a comparison of the neural network trained with data in time versus the same data expressed in the frequency domain is obtained, seen from the increase in performance and the percentage of gesture detection.**

***Keywords*—Neural networks; electromyographic signals; Myo armband; tensorflow; fast fourier transform**

I. Introduction

Currently, there are different types of human-machine interfaces [1] (HMI) developed for applications in areas like automation [2], robotics [3], biomedicine [4], biometrics [5], among others. That has led to users needing specialized studies, knowledge, or skills on information technology to implement and operate such applications properly. For this reason, it is becoming important to minimize the complexity of these types of controls, design more straightforward and more intuitive human-machine interfaces, which take advantage of the benefits of human biomechanics, and make it easy and safer handling of the applications. Therefore, the aim is to have control interfaces that do not require much prior training and are as natural as possible, reaching non-invasive devices which can be used as a clothing accessory without needing the help of external personnel for configuration or startup.

For years one of the techniques that have been used for the development of HMI is electromyographic (EMG) signals [6]. The EMG signals measure the electrical currents generated in the muscles during their contraction, representing neuromuscular activities [7]. An example of this is interfaces that detect the gestures of a human forearm, which use the electrical impulses of the forearm to control machinery, robots [8], prosthesis [9], [10], home automation systems [11], [12], personal identification systems [13], [14], IoT systems [15], among others.

On the other hand, a series of transformations have been used in the preprocessing of EMG signals for a couple of decades, taking into account problems present in periodicity, frequency behavior, stationary behavior, and fast transient behavior. Consequently, EMG signals have been acquired in the time domain, and mathematical transformations have been used to bring them to the frequency domain [16]. For example, Wavelet transform (WT) and the fast Fourier transform (FFT) are primary tools for analyzing and subsequent use of these signals.

Additionally, in recent years the use of Artificial Neural Networks (ANN) has become extensive for the classification of EMG signals [17], [18]. Due to the complexity of analyzing the intrinsic characteristics of these signals in terms of variance identification, average, length, zero crossing, median, and frequency, as to propose an algorithm. Thus, some previous works have focused on acquiring a database of signals in the time domain to use them as a knowledge base to train an ANN-based machine learning system [19], [20]. The amount of information to be analyzed becomes significant due to pure or "raw" EMG signals in the time domain.

Therefore, the objective of this research is to develop an algorithm in Python language that detects the gestures of a human forearm, using the EMG signals of this part of the body, to make a direct comparison between the use of these pure signals in time and the same signals transformed to the frequency domain through an FFT. These last to training and subsequent verification of an ANN and validate the algorithm's performance.

The paper is organized as follows: Section 2 presents the methodology proposed to detect EMG signals, create a dataset, preprocess the data in the time domain and get through FFT the data in the frequency domain, and design ANN to training and classification arm gestures. Section 3 presents the results of implementing the ANN algorithm in Python language, testing, and evaluating its performance. Section 4 presents the conclusions about this research's main ideas, including possible future jobs.

II. Methodology

A. Data Acquisition System

Technological advances in miniaturization and high performance of electronic devices have allowed advances in biomedicine, applied to human-machine interfaces, in this case, the use of wearable devices, particularly the Myo armband handle designed by the company Thalmic labs inc [21]. This armband is equipped with eight EMG electrodes, a 3-axis gyroscope, a 3-axis accelerometer, and a 3-axis magnetometer to perform IMU metrics. The EMG electrodes on the handle detect signals related to the muscular activities of the user's forearm, and the IMU detects forearm movements in three-dimensional space. This data acquired by the handle is sent via Bluetooth Low Energy (BLE), which allows a 3D reconstruction of human forearm movements, making it a good choice for this type of human-machine interface [22].

The Myo armband was chosen for this work as the EMG data acquisition system, running on Windows operating system using Myo Connect drivers as a base. Moreover, TensorFlow 'TFF' machine learning libraries were used to design the ANN in the software part.

As a first step to create a dataset, data from a human forearm was acquired as a control interface using the Myo™ Gesture Control armband tool. The signals were captured using Python scripts to read the data and store them in flat format. Then, those EMG signals were analyzed to be used as a knowledge base for the training and verification of the ANN-based learning system. For this purpose, it was chosen five basic gestures made with the forearm, as shown in Fig. 1. Each gesture is captured by the Myo armband device for a time of one second, with a sampling frequency of 200Hz. These signals are considered a significant sample of gesture behavior. The data capture was performed through a Python script, which was in charge of measuring the time, completing the capture, and finally writing a flat file.

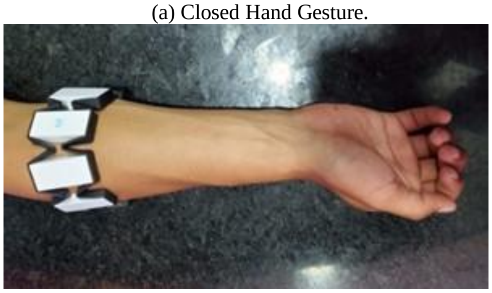
B. Characteristics of Artificial Neural Network

In recent years, ANNs have been used as a basis for recognition and classification tasks [23], [24]. Like in this case, it is required to associate patterns from a previously generated dataset, identifying from each muscular gesture performed by the arm those relevant characteristics that make it different from another gesture. It has been decided to use an ANN for this task because:

- it has the natural ability to acquire knowledge through experience,
- it can be easily adapted dynamically depending on the learning environment, and
- it has a high level of fault tolerance, supporting missing input data or significant damage to its structure and continuing with a good performance.

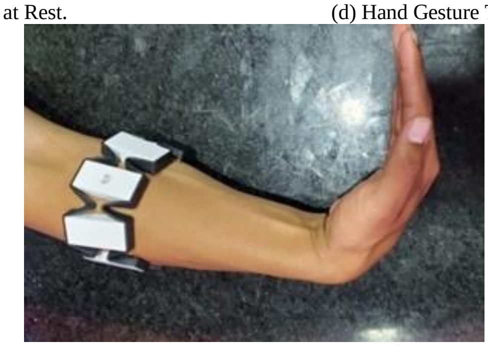
(a) Closed Hand Gesture.

(b) Open Hand Gesture.



(c) Hand Gesture at Rest.

(d) Hand Gesture Turned to the Right.



(e) Hand Gesture Turned to the Left.

Fig. 1. Basic Gestures Captured with the Myo Armband.

So, the first methodological step was to collect the data, with specific characteristics, with a minimum amount of data, and in different individuals to have an adequate knowledge base. After that, it is necessary to decide which libraries must implement the learning system. Given its characteristics and ease of implementation, it chose to develop ANN using a library called TensorFlow Federated "TFF" developed by Google. It is an open-source framework for programming code for machine learning with decentralized data.

This library allows different architectures to be proposed in terms of the type of training, the platform on which they run and the different programming languages used, and the devices on which the final application will run. The architecture of TFF makes it possible to run on different platforms, i.e., CPUs, GPUs, and on a PC or a mobile device, in addition to being compatible with different programming languages such as Java, JavaScript, C, Go, Python, among others. However, it has been evidenced that the language most used to implement TFF is Python, so it is the one that will be used for the development of the algorithm to model the ANN.

After choosing the architecture, the device, and the programming language, it is necessary to define the size and characteristics of the dataset in terms of:

- the type of file to work with,
- the amount of data to be used for training, and
- the amount of data to be used to verify the learning performed.

Once it is clear if the dataset is local, built from its own data captures, the next step is to define the architecture of the ANN in terms of the number of neurons in the input and output layers, the number of hidden layers, and therefore the number of neurons used for the process. In addition to which activation function is indicated for the type of data being used.

Depending on the hardware characteristics, which in this case are local, using a CPU in a desktop PC, the performance of the neural network training software must be taken into account as a limiting parameter for the construction of the ANN model. In other words, the designed architecture and the processing time required must be considered to perform the training process. For this case, being local processing using a CPU requires a certain number of hours and does not allow significant changes to the neural network architecture in an agile way.

Finally, having the internal architecture of the ANN, it is necessary to perform learning tests, error quantity and verify if the system learned, for which TFF offers reports and graphs for error verification ROC curves, as well as verification of system learning through confusion matrices.

C. *Block Diagram - Proposed Overall Solution*

After defining the work methodology with which the solution will be designed and considering that such a scheme applies to different work scenarios, a general flow diagram is proposed in Fig. 2 to reach a system that detects gestures in a usable knowledge base for different applications. On the other hand, it is essential to emphasize that it seeks to compare the data input, firstly the data as captured with the bracelet, in the time domain, simply with a normalization process, all this compared with a preprocessing of the input signals brought into a frequency domain.

D. *Pre-process Time or FFT*

Sixty-six thousand four hundred seventy-nine (66479) signals with a duration of one second were captured, corresponding to five different gestures using the Myo armband performed by three different individuals, as shown in Fig. 1. Subsequently, EMG signals of this dataset are preprocessed before being stored as local files, modeled, and classified to be usable for ANN training.

Fig. 3 shows a signals capture of the 8 EMG channels for a gesture, with a duration of one second and a sampling frequency of 200 Hz; it is possible to observe the characteristics of the signals captured in the time domain by the Myo armband. For this particular case, the signal is between a range of -50 and 50 units; it is evident that the most significant part of the signal is in its first 64 samples, in which the most substantial part of the gesture is found. For this reason, the ANN has an input layer of 64 neurons.

Fig. 4 shows the samples transformed to the frequency domain. A preprocessing of the EMG signals is performed, transforming those to the frequency domain, using the fast Fourier transform specifically. As base parameters for the Fourier transform, it used the criteria of twice the sampling frequency of the signal, i. e., the base frequency of the transform was performed at 400Hz. Also, the data of the magnitude and bilateral angle of the signal centered at zero were taken.

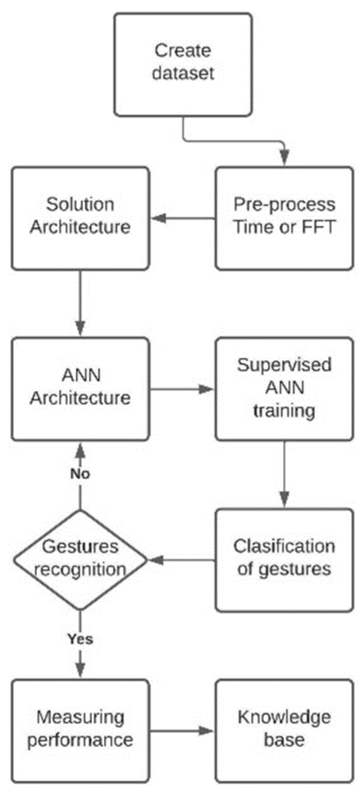


Fig. 2. Flow Chart of the Proposed Overall Solution.

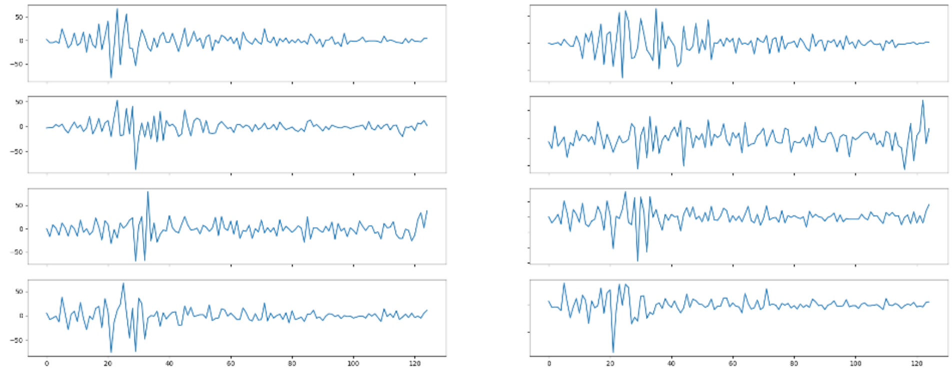


Fig. 3. EMG Signals Captured with the Myo Armband of the Left-hand Gesture in the Time Domain

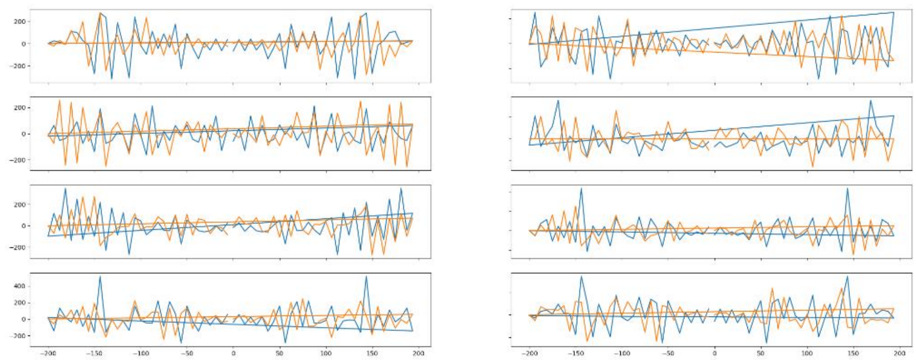


Fig. 4. EMG Signals Captured with the Myo Armband of the Left-hand Gesture in the Frequency Domain.

E. Architecture of Artificial Neural Network

Regarding the architecture of the solution, it was chosen to perform the data capture and network training process in a supervised manner using a local PC with the following hardware configuration: 5800x processor, RTX2070 GPU, 32Gb RAM at 3600Mhz, and the following software versions were used: Python version 3.9 and TensorFlow version 2.7.0.

Regarding the ANN creation and testing process, the first step is to characterize the data set that will be entered. Then, tests are performed with different numbers of training epochs or iterations to different configurations of the layers in the designed multilayer network. Subsequently, the architecture of the ANN must be defined, as shown in Fig. 5. The input layer has sixty-four neurons, that number of neurons taking into account the size of the dataset samples. The hidden layers have a number of neurons variable depending on the minimum performance required for this application; in Fig. 5. the change in the number of neurons is represented as N. The output layer has five neurons, that number of neurons considering the different gestures to be identified.

When talking about ANNs, it is necessary to have a minimum model to recalculate the weights that will model the behavior of the network. In this case, a simple backpropagation model is chosen that complies with the following equations,

Input layer: $^{(i)}$ (1)

Hidden layer: $a_1^{(i)} = \sigma(W_1 x^{(i)} + b_1)$ (2)

Output layer: $\hat{y}^{(i)} = \sigma(W_2 a_1^{(i)} + b_2)$ (3)

Where $^{(i)}$ is the input, the captures that were made of the three individuals performing the five gestures, W_1, b_1, W_2, b_2 are the matrices of weights and vectors of independent values used in the layers of (1) and (2), which are initialized randomly. The nonlinear activation function is σ. The result in (3) is represented by $\hat{y}^{(i)}$ where i is the desired output estimate [25].

Once it has a tentative ANN architecture, it is performed a series of tests with different network models until it achieves accuracy percentages greater than 80% and an amount of lost data less than 20%. Table I shows the different models of the ANN, taking as a knowledge base the data in the time domain.

The third ANN model uses an architecture with 66479 neurons in the first hidden layer and 300 neurons in the second hidden layer due to an activation method called RELU. RELU is a function that allows data to pass through or not to the next layer, depending on the result of the neuron weighting with equation (2). If the result of the neuron weighting is negative or zero, it does not pass to the next layer; otherwise, any positive number passes to the next layer. (Tf.Nn.Relu | TensorFlow Core v2.7.0, n.d.). Moreover, the first layer of the ANN has a dropout of 20%, which consists of randomly establishing in each iteration of training which neuron should be deactivated. The number of deactivated neurons in the layer will depend on the percentage entered into the dropout method. That is done not to overtrain the ANN. (Tf.Keras.Layers.Dropout | TensorFlow Core v2.7.0, n.d.).

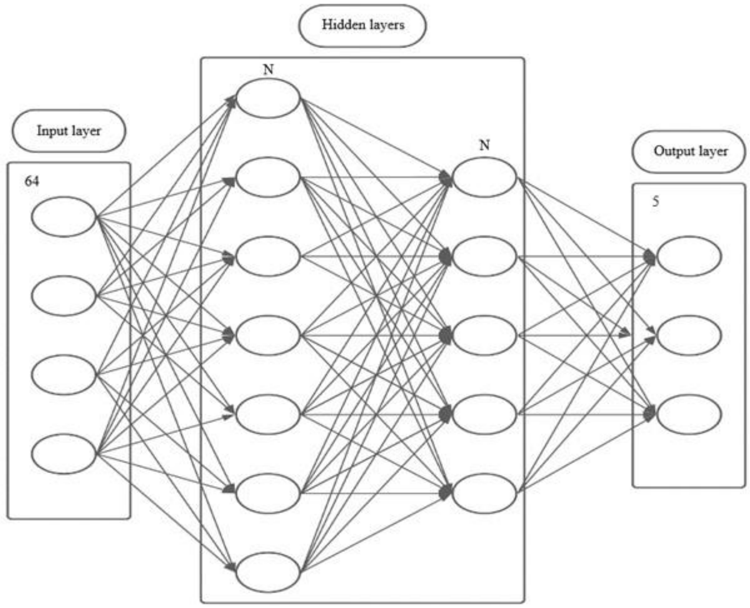


Fig. 5. Graphical Model of an ANN

TABLE I. DIFFERENT MODELS OF THE INTERNAL ARCHITECTURE OF THE ANN, USING THE DATA IN TIME

Model	Epochs	Neurons hidden layer 1	Neurons hidden layer 2	Neurons hidden layer 3	Accuracy	Lost
1	50	15000	100	N/A	20%	30%
2	2000	15000	100	20	25%	40%
3	1000	66479	300	100	90%	0,80%

TABLE II. MODEL OF THE INTERNAL ARCHITECTURE OF THE ANN, USING THE DATA AT FREQUENCY

Model	Epochs	Neurons hidden layer 1	Neurons hidden layer 2	Neurons hidden layer 3	Accuracy	Lost
Unique	1000	66479	300	100	95%	0,87%

When transforming the data to the frequency domain using FFT, it decreases the number of samples needed and minimizes some problems that the signals present in the time domain; in this case, it has the following ANN model, as shown in Table II.

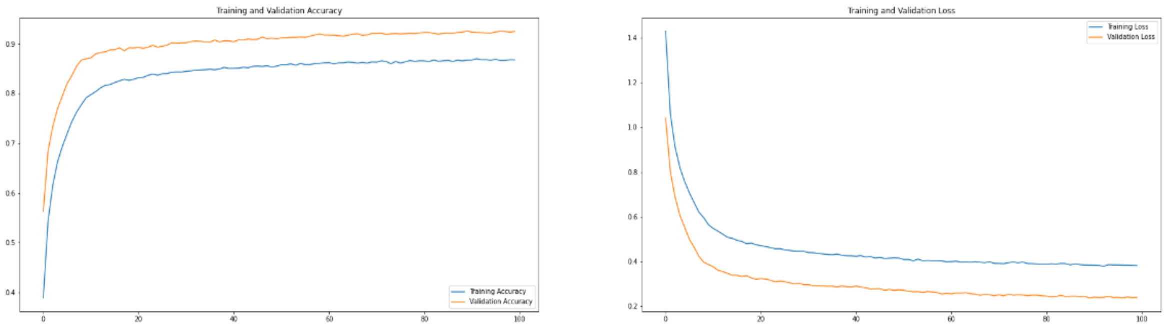
F. Measuring Performance of Artificial Neural Network

Once the ANN model has been found, the parameters considered to verify the correct operation of the system are its performance and margin of error, in addition to analyzing possible training errors. This process has been done two previous times, modifying the data set and remodeling the ANN architecture, either by changing the weight of the neurons proportionally to the error or by reducing or increasing the layers of neurons to reduce overtraining or underfitting. When this architecture is already defined, the behavior of the error and performance of the network is analyzed. The results for the third model are shown in Fig. 6; such data analysis is performed for the ANN with the input data in the time domain.

It can be seen that the probability of getting the gesture right is approximately 92%, and there is a loss of less than 0.4%; these results were obtained with the data set of 66479 captures or samples of the input signal in the time domain.

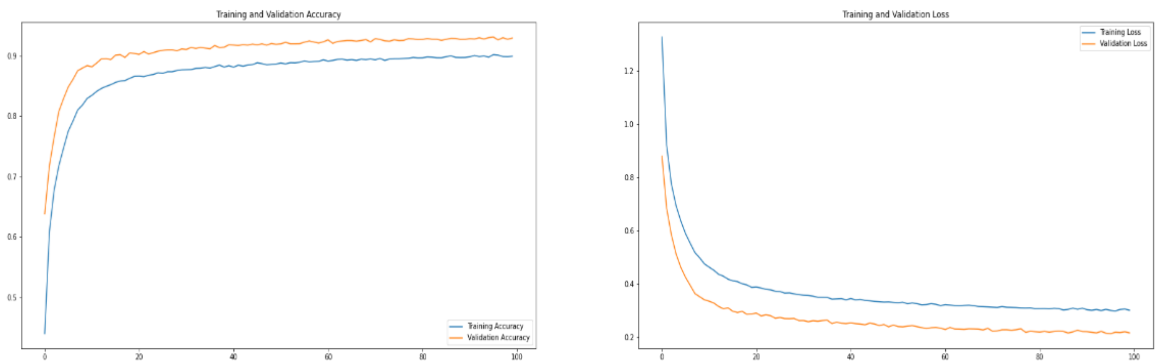
In the same way, tests were performed for the case of the chosen model, using the data in the frequency domain, such validation measurement of accuracy parameters and data loss, are shown in Fig. 7.

It can be seen that the probability of getting the gesture right is approximately 93%, and there is a loss of less than 0.21%; these results were obtained with the data set of 66479 captures or samples of the input signal in the time domain transform to the frequency domain.



(a) Accuracy Validation. (b) ANN Model Loss.

Fig. 6. Measurement of Training Parameters for Third Model with Time Domain Input Data.



(a) Accuracy Validation. (b) ANN Model Loss.

Fig. 7. Measurement of Training Parameters with Frequency Domain Input Data.

III. Results

The ANN training yielded a data loss of 24.76% and an accuracy rate of 91.98%, with a thousand epochs in the training of the ANN data, data shown in Fig. 8, all this for the time domain data.

The ANN training yielded a 21.66% data loss and a 92.95% accuracy percentage, with a thousand epochs in the training of the ANN data, data shown in Fig. 9, all this for the data in the frequency domain.

One of the most used tools to validate the efficiency of the selected neural network model is the confusion matrix of the model, which is represented in a graph. This graph shows a matrix that shows the percentage of accuracy of the ANN when predicting a gesture, and it can also see the percentage of error when predicting another gesture that does not correspond to the one entered.

In order to appreciate the accuracy of the model, it is necessary to consider the diagonal of the matrix that begins in the upper left corner and ends in the lower right corner. The numbers that make up the diagonal indicate the percentage of accuracy in predicting the gesture correctly; the other fields of the matrix are the percentage of error in predicting the gesture. Fig. 10 shows the graphs of the confusion matrix for the time domain and frequency domain of the ANN model developed.

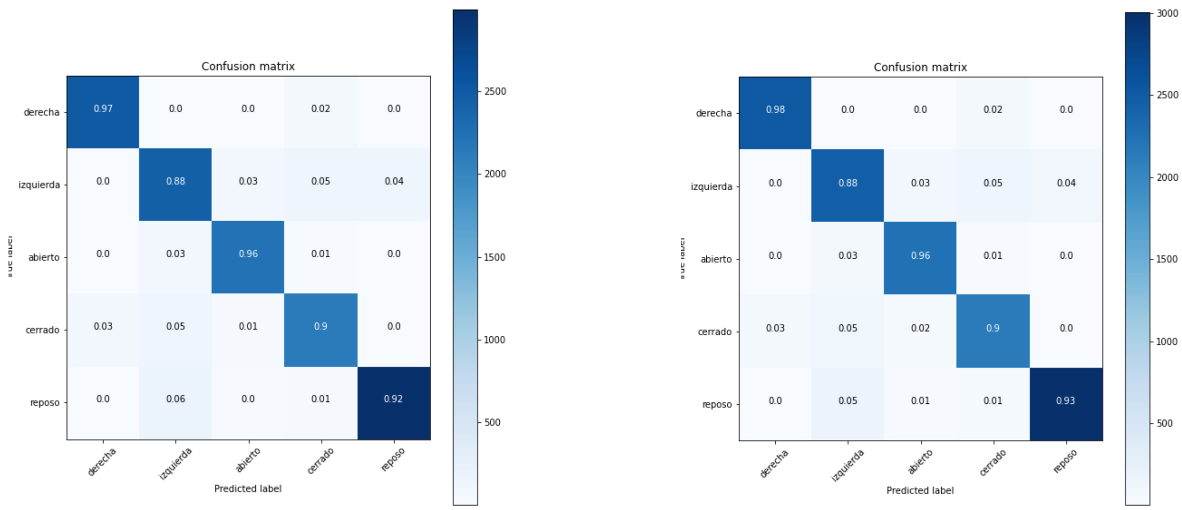
It can observe in confusion matrices that despite the pre-processing of the information, using the fast Fourier transform, the performance only improves by 1% for some gestures. These matrices clearly show that the system successfully classified the gestures, both in the time and frequency domains, consistently achieving prediction percentages higher than 88%.

```
831/831 - 3s - loss: 0.3864 - accuracy: 0.8656 - val_loss: 0.2430 - val_accuracy: 0.9220 - 3s/epoch - 4ms/step
Epoch 99/100
831/831 - 3s - loss: 0.3903 - accuracy: 0.8650 - val_loss: 0.2427 - val_accuracy: 0.9220 - 3s/epoch - 3ms/step
Epoch 100/100
831/831 - 3s - loss: 0.3881 - accuracy: 0.8653 - val_loss: 0.2476 - val_accuracy: 0.9198 - 3s/epoch - 3ms/step
```

Fig. 8. Test Values Obtained in ANN Training in the Time Domain.

```
831/831 - 3s - loss: 0.3044 - accuracy: 0.8989 - val_loss: 0.2178 - val_accuracy: 0.9301 - 3s/epoch - 3ms/step
Epoch 99/100
831/831 - 3s - loss: 0.3060 - accuracy: 0.8986 - val_loss: 0.2207 - val_accuracy: 0.9276 - 3s/epoch - 4ms/step
Epoch 100/100
831/831 - 3s - loss: 0.3015 - accuracy: 0.8994 - val_loss: 0.2166 - val_accuracy: 0.9295 - 3s/epoch - 4ms/step
```

Fig. 9. Test Values Obtained in the ANN Training in the Frequency Domain.



(a) Confusion Matrix for Time Domain.

(b) Confusion Matrix for the Frequency Domain.

Fig. 10. Confusion Matrices of the ANN Model.

IV. Conclusion

This paper documents the training process of an ANN to detect five gestures captured from a human forearm of 3 different individuals, focusing on the differences between capturing the "raw" data in the time domain versus pre-processing them by transforming them to the frequency domain. It was verified that transforming dataset samples to the frequency domain removes some time-domain dataset issues such as delays, level offsets, and signal offsets. Also, it was possible to reduce the number of samples necessary to generate the minimal identification information of the gestures.

In some previous literature, a significant improvement was observed in working the systems in the frequency domain, avoiding problems inherent to the EMG signals. For this reason, it was expected that there would be an improvement in the detection process of the gestures. However, the increase in the measurement parameters and verification of the ANN has only reflected improvements close to 1%.

As future work, it is proposed to improve the data capture process, increasing the signal's sampling period over time to have more detectable harmonics in the frequency domain and have more elements to perform the training process of the ANN. Having a new sampling rate would increase the number of neurons in the input layer, and therefore the rest of the hidden capabilities would have to be modified. Finally, it would have to alter the epochs in the training process, having more samples of different individuals to have a complete dataset, looking for an improvement that exceeds the threshold of 90% detection of gestures.

Acknowledgment

The Universidad Distrital Francisco José de Caldas supports this work, with the research group belonging to the Technological Faculty; specifically, the research group SIE (Embedded Informatics Security). Currently, the work is focused on the digital processing of signals, images, and intelligent algorithms that seek to be applied in stand-alone solutions.

References

[1] M. A. Razzaq, M. A. Qureshi, K. H. Memon, S. Ullah, and R. Y. Khan, “A survey on user interfaces for interaction with human and machines,” 2017. [Online]. Available: https://blog.mozilla.org/labs/2007/07/the-graphical-keyboard-user.

[2] H. Oliff, Y. Liu, M. Kumar, and M. Williams, “A framework of integrating knowledge of human factors to facilitate hmi and collaboration in intelligent manufacturing,” in Procedia CIRP, 2018, vol. 72, pp. 135–140. doi: 10.1016/j.procir.2018.03.047.

[3] A. Grabowski, J. Jankowski, and M. Wodzyński, “Teleoperated mobile robot with two arms: the influence of a human-machine interface, VR training and operator age,” International Journal of Human Computer Studies, vol. 156, Dec. 2021, doi: 10.1016/j.ijhcs.2021.102707.

[4] S. P. Zavala, S. G. Yoo, and D. E. Valdivieso Tituana, “Controlling a wheelchair using a brain computer interface based on user controlled eye blinks.” [Online]. Available: www.ijacsa.thesai.org.

[5] N. Belgacem, R. Fournier, A. Nait-Ali, and F. Bereksi-Reguig, “A novel biometric authentication approach using ECG and EMG signals,” Journal of Medical Engineering and Technology, vol. 39, no. 4, pp. 226–238, May 2015, doi: 10.3109/03091902.2015.1021429.

[6] A. Patel, J. Ramsay, M. Imtiaz and Y. Lu, "EMG-based human machine interface control," 2019 12th International Conference on Human System Interaction (HSI), 2019, pp. 127-131, doi: 10.1109/HSI47298.2019.8942598.

[7] M. B. I. Reaz, M. S. Hussain, and F. Mohd-Yasin, “Techniques of EMG signal analysis: detection, processing, classification and applications,” Biological Procedures Online, vol. 8, no. 1, pp. 11–35, Mar. 2006, doi: 10.1251/bpo115.

[8] S. Gowtham, K. M. A. Krishna, T. Srinivas, R. G. P. Raj and A. Joshuva, "EMG-based control of a 5 DOF robotic manipulator," 2020 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET), 2020, pp. 52-57, doi: 10.1109/WiSPNET48689.2020.9198439.

[9] A. F. Ruiz-Olaya, C. A. Quinayas Burgos and L. T. Londoño, "A low-cost arm robotic platform based on myoelectric control for rehabilitation engineering," 2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), 2019, pp. 0929-0933, doi: 10.1109/UEMCON47517.2019.8993080.

[10] J. Vogel et al., “EDAN: An EMG-controlled daily assistant to help people with physical disabilities,” in IEEE International Conference on Intelligent Robots and Systems, Oct. 2020, pp. 4183–4190. doi: 10.1109/IROS45743.2020.9341156.

[11] P. J. Gonzalo and A. Holgado-Terriza Juan, “Control of home devices based on hand gestures,” in 5th IEEE International Conference on Consumer Electronics - Berlin, ICCE-Berlin 2015, Jan. 2016, pp. 510–514. doi: 10.1109/ICCE-Berlin.2015.7391325.

[12] Y. Muranaka, M. Al-Sada, and T. Nakajima, “A home appliance control system with hand gesture based on pose estimation,” in 2020 IEEE 9th Global Conference on Consumer Electronics, GCCE 2020, Oct. 2020, pp. 752–755. doi: 10.1109/GCCE50665.2020.9291877.

[13] L. Lu, J. Mao, W. Wang, G. Ding, and Z. Zhang, “A study of personal recognition method based on EMG signal,” IEEE Transactions on Biomedical Circuits and Systems, vol. 14, no. 4, pp. 681–691, Aug. 2020, doi: 10.1109/TBCAS.2020.3005148.

[14] J. S. Kim and S. B. Pan, “A Study on EMG-based Biometrics.”

[15] M. Nguyen, T. N. Gia, and T. Westerlund, “EMG-based IoT system using hand gestures for remote control applications,” in 7th IEEE World Forum on Internet of Things, WF-IoT 2021, Jun. 2021, pp. 911–912. doi: 10.1109/WF-IoT51360.2021.9595957.

[16] L. Lu, J. Mao, W. Wang, G. Ding and Z. Zhang, "An EMG-based personal identification method using continuous wavelet transform and convolutional neural networks," 2019 IEEE Biomedical Circuits and Systems Conference (BioCAS), 2019, pp. 1-4, doi: 10.1109/BIOCAS.2019.8919230.

[17] R. Shioji, S. Ito, M. Ito and M. Fukumi, "Personal authentication and hand motion recognition based on wrist EMG analysis by a convolutional neural network," 2018 Joint 10th International Conference on Soft Computing and Intelligent Systems (SCIS) and 19th International Symposium on Advanced Intelligent Systems (ISIS), 2018, pp. 1172-1176, doi: 10.1109/SCIS-ISIS.2018.00184.

[18] S. Shin, J. Jung and Y. T. Kim, "A study of an EMG-based authentication algorithm using an artificial neural network," 2017 IEEE SENSORS, 2017, pp. 1-3, doi: 10.1109/ICSENS.2017.8234158.

[19] T. Wilaiprasitporn, A. Ditthapron, K. Matchaparn, T. Tongbuasirilai, N. Banluesombatkul, and E. Chuangsuwanich, “Affective EEG-based person identification using the deep learning approach,” IEEE Transactions on Cognitive and Developmental Systems, vol. 12, no. 3, pp. 486–496, Sep. 2020, doi: 10.1109/TCDS.2019.2924648.

[20] U. 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, Apr. 2019, doi: 10.1109/TNSRE.2019.2896269.

[21] P. Visconti, F. Gaetani, G. A. Zappatore, and P. Primiceri, “Technical features and functionalities of Myo armband: an overview on related literature and advanced applications of myoelectric armbands mainly focused on arm prostheses,” International Journal on Smart Sensing and Intelligent Systems, vol. 11, no. 1, pp. 1–25, 2018, doi: 10.21307/ijssis-2018-005.

[22] T. Phienthrakul, “Armband gesture recognition on electromyography signal for virtual control,” in 2018 10th International Conference on Knowledge and Smart Technology: Cybernetics in the Next Decades, KST 2018, Aug. 2018, pp. 149–153. doi: 10.1109/KST.2018.8426118.

[23] J. Xu, T. Li, Y. Chen and W. Chen, "Personal identification by convolutional neural network with ECG signal," 2018 International Conference on Information and Communication Technology Convergence (ICTC), 2018, pp. 559-563, doi: 10.1109/ICTC.2018.8539 632.

[24] M. F. Guo, X. D. Zeng, D. Y. Chen, and N. C. Yang, “Deep-learning-based earth fault detection using continuous wavelet transform and convolutional neural network in resonant grounding distribution systems,” IEEE Sensors Journal, vol. 18, no. 3, pp. 1291–1300, Feb. 2018, doi: 10.1109/JSEN.2017.2776238.

[25] E. H. Galvis-Serrano, I. Sánchez-Galvis, N. Flórez, and S. Zabala-Vargas, “Classification of gestures of the colombian sign language from the analysis of electromyographic signals using artificial neural networks,” Informacion Tecnologica, vol. 30, no. 2, pp. 171–179, Mar. 2019, doi: 10.4067/S0718-07642019000200171.