

## CNN Architectures for Hand Gesture Recognition using EMG Signals Through Wavelet Feature Extraction

Natalie Segura Velandia, Robinson Jimenez Moreno and Astrid Rubiano  
Department of Mechatronics, Faculty of Engineering, Nueva Granada Military University,  
Bogota, Colombia

---

**Abstract:** This study presents the implementation of 3 convolutional neural network architectures for the recognition of hand gestures by means of electromyographic signals. The acquisition of signals is done by means of electrodes located in the forearm and the development platform specialized in biomedical signals MySignals HW V2.0 which will be applied a pre-processing of the signal by means of the Wavelet Packet Transform (WPT) for the feature extraction. The architectures that are proposed have as input base to the network the map of features obtained by the wavelet power spectrum with which the database of training and validation was constructed. Finally, in the tests performed in real time, the first architecture reached an accuracy of 93.8325%, the second architecture, reaches a degree of accuracy of 95.8824% and finally, the third architecture reaches an accuracy of 96.4706%. This means that the architecture with the highest accuracy performs better when it comes to recognizing gestures, even with small databases.

**Key words:** Deep convolutional neural network, EMG signal, wavelet power spectrum, discrete wavelet transform, gesture recognition, validation

---

### INTRODUCTION

Electromyography (EMG) signals have been the subject of study after the discovery of diseases caused by the neuromuscular system. Therefore, an EMG signal provides the information necessary to diagnose a neuromuscular disease or some damage caused by an injury to any of the muscles of the body. However, researchers and scientists have taken advantage of these signals by implementing them for the control of prostheses (Cipriani *et al.*, 2008; Romo *et al.*, 2007) and robotic manipulators (Fukuda *et al.*, 2003). In other words, they are electrical signals that are generated by the various ( $\text{Na}^+$ ,  $\text{K}^+$ ,  $\text{Ca}^{++}$ ,  $\text{Cl}^-$ , etc.) ions (Cardinali, 2007) that are present in the muscle during the flexion and contraction movements which can be monitored by 2 techniques. The first technique is invasive through electrodes inserted inside the muscle, mainly used for the study of deep muscles which consists of stimulating the muscle with small electrical impulses and observing the behavior of the muscle before the discharges. The second is through non-invasive electrodes, using surface electrodes that allow the study of muscle activity in dynamic actions in other words it allows simultaneous analysis of different muscles in movements.

Monitoring of EMG signals then allows discrimination of muscle failures but also characterization of intention of movement of a person. For this it is necessary to use pattern recognition techniques, framed

within Artificial Intelligence (AI) algorithms, currently these have been focused on deep neural networks which correspond to a model inspired by the functioning of the brain which analyzes information as a deep architecture, for example, 5-10 layers of processing for visual information (Serre *et al.*, 2007). By its layered learning architecture, this technique allows to reach several levels of abstraction and to obtain remarkable results in the recognition of patterns (LeCun *et al.*, 2015). A specific algorithm of deep learning are the Convolutional Neural Networks (CNN) defined as multi-layered, trainable structures which guarantee with their convolution filters a learning of specific characteristics during each level of the network.

Recently, the development of applications to create human-machine interfaces, presents reductions in the performance of the results, due to a poor implementation in the process of extraction of characteristics of the corporal signals. Because of this in Englehart *et al.* (2001) they focus an investigation on the recognition and classifications of EMG signal patterns, the technique is based on the implementation of the wavelets functions for the analysis of the main characteristics of the signal, the results show that the myoelectric control has a greater precision when four channels are implemented, however, the increase of channels could cause delays in the response.

On the other hand, the investigations focused on the processing of EMG signals using CNN for the recognition

of patterns, just begin to be explored. By Park and Lee (2016), a decoding method adaptable to each user to obtain the intention of movement is proposed using EMG signals for the portable rehabilitation of upper limbs where they use a CNN network to predict 6 different hand movements. By Atzori *et al.* (2016), they use a simple CNN architecture, consisting of 5 convolution layers where they conclude that the architecture is successful, although with an accuracy of  $66.59 \pm 6.4\%$ , due to the large number of patterns that they seek to recognize.

Taking into account the above, the contribution of the present study is to evaluate 3 CNN architectures for the recognition of gestures, implementing a technique specialized in feature extraction with the purpose of improving the input of the network and in this way and in this way, to improve the precision in the recognition of the network.

## MATERIALS AND METHODS

The surface Electromyography signals (EMGs) are one-dimensional patterns, so, any technique of extraction features is applicable to this type of signal (Romo *et al.*, 2007). The acquisition of the signals is performed using the MySignals HW V2 card (Anonymous, 2017), obtaining a continuous signal that expresses the behavior of the muscle by the voltage variation before a movement as a function of time.

Thus, for the development of an algorithm capable of recognizing hand gestures based on EMG signals, the electrodes should be placed in the muscles where hand movements predominate, using the methodology presented in Fig. 1. Figure 1 shows the steps of capture of the signal its digitalization to a computer, processing by means of MATLAB for feature extraction and the training of the convolutional neural network evaluated by means of confusion matrices. These steps are described as follows.

**Signal acquisition:** The EMG signals are generated by the muscular contraction which leads to an analysis of the muscular region involved in the movements caused by the gestures to be classified. Derived from this, the short radial extensor muscle of the carpus is chosen as shown in Fig. 2a. Therefore, it is necessary to make a correct placement of the electrodes as shown in Fig. 2b and it must have a reference electrode or ground which should be located in a place where it is not affected by the gestures as it is presented in Fig. 2c. It is important to emphasize that the EMGs signals are collected by non-invasive surface electrodes, made up of an electrolytic gel layer responsible for increasing the conductivity and achieving a better adhesion on the skin.

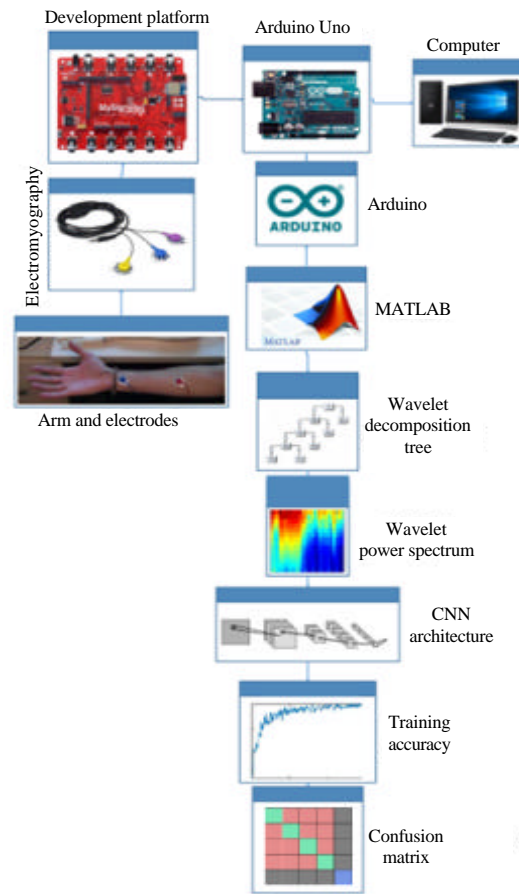


Fig. 1: Blocks diagram of operation of the algorithm developed

**Hand gesture recognition:** According to the muscular group chosen, a group of gestures are set that will be recognized and classified. A previous analysis of these is important to establish which are the movements associated to the location of the electrodes which will allow a better extraction of characteristics. Throughout this work, the gestures presented in Fig. 3 are used.

**Signal preprocessing:** For the analysis of EMG signals it is necessary to perform a pre-processing to eliminate noise and to be able to extract the real characteristics of the same for its classification (Romo *et al.*, 2007). By Phinyomark *et al.* (2011) a case of success in the pre-processing stage is presented, applying the wavelet denoising algorithm technique which allows to achieve an increase in the precision of the recognition of EMG signals between 50 and 60%. Next, the basic concepts of this technique are presented which are basis for the suppression of noise in this research.

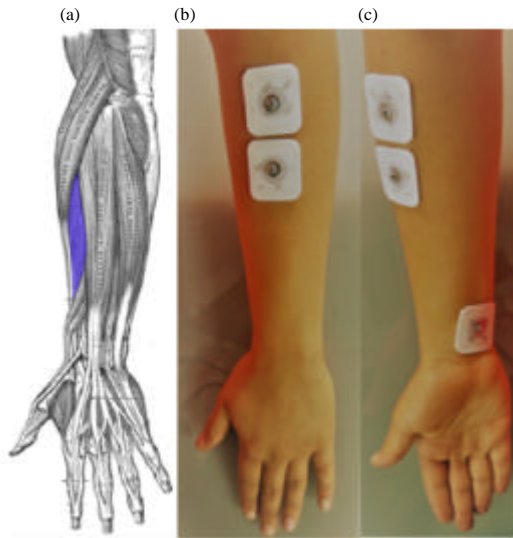


Fig. 2: Location of the electrodes on the forearm: a) The posterior region of the forearm is seen where the short radial extensor muscle of the carpus is illustrated; b) The location of the 2 electrodes that acquire the signal is shown and c) The third electrode is shown

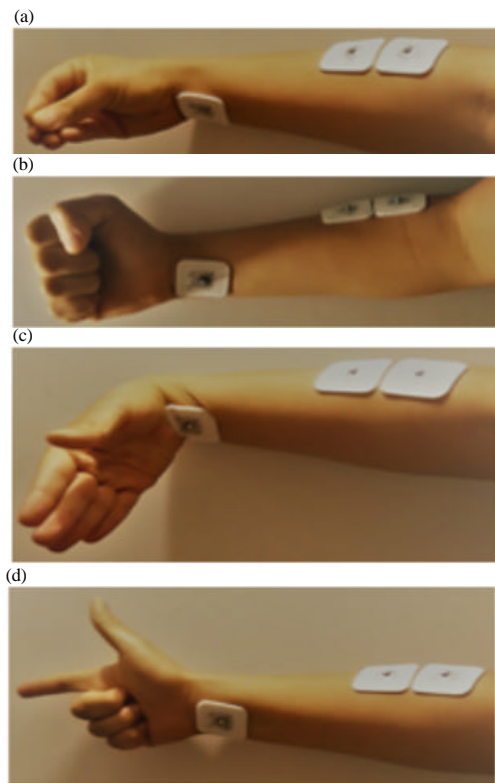


Fig. 3: Hand gestures: a) "Close-fingers" gesture; b) "Fist" gesture; c) "Wave-in" gesture and d) "Gun" gesture

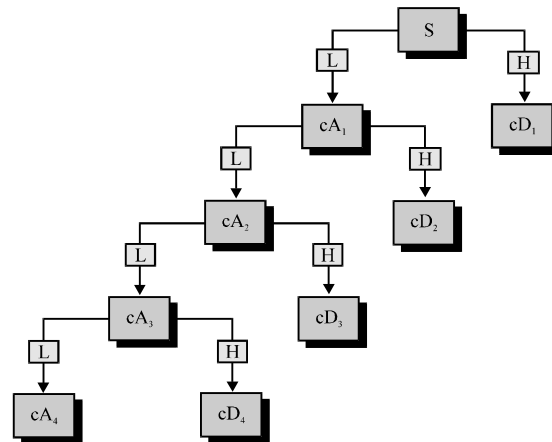


Fig. 4: Wavelet decomposition tree (Phinyomark *et al.*, 2011) where S represents the Signal input, L the Low pass filter, H the High pass filter, cA Approximation coefficient and cD Detail coefficient

#### Feature extraction using discrete wavelet transform

**Wavelet Packet Transform (WPT):** The WPT is developed by a modification of the Wavelet Transform (WT) responsible only to decompose the approximations (Low frequencies) of the signal. In other words, if the most important features of the signal are in the details (high frequencies) this method could cause loss of information (Shinde and Hou, 2005). The WPT consists of passing the signal through several levels which is recommended to determine by means of a signal behavior branch (Misiti *et al.*, 2017).

It is important to emphasize that in the first level the approach obtains new details and approximations as presented in Fig. 4, thus, achieving a better resolution in frequency which in turn allows to increase the quality of the features of the signal (Misiti *et al.*, 2017).

**Mother wavelet function:** The mother wavelet is a necessary function to implement the WPT its choice is made identifying the best correlation that it has with the signal under study. It is associated with the mother wavelet function Daubechies (dB) (Daubechies, 1992) which is defined from dB 1-45. This order is determined by the behavior of the function, for example, Fig. 5 shows the behavior from dB 2-10. This function is mainly applied in low order configuration (dB 1-20) (Wu and Liu, 2008). According to the above, the Daubechies function is chosen with an order 7 (dB 7), according to the correlation with the EMG signal that it has.

**Wavelet packet spectrum:** This package is responsible for processing the coefficients provided by the WPT where

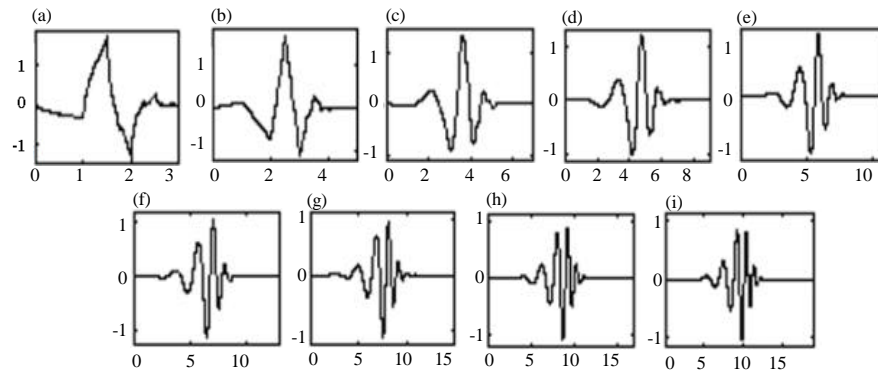


Fig. 5: Wavelet “Daubechies” family behavior: a) dB 2; b) dB 3; c) dB 4; d) dB 5; e) dB 6; f) dB 7; g) dB 8; h) dB 9 and i) dB 10

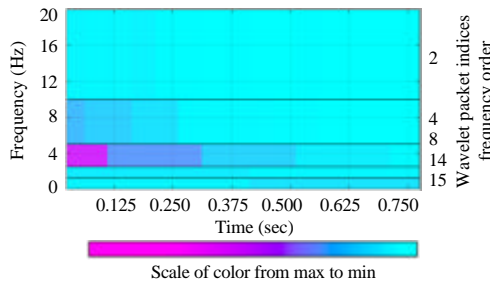


Fig. 6: Wavelet packet decomposition “Packet spectrum”

it performs the following processes: first, extracts the coefficients from the WPT terminals and converts them to absolute values. Second, it determines the length of time corresponding to each wavelet packet coefficient and it fills the time slots between neighbors by creating a vector of length equal to the first level of the wavelet packet tree object (Misiti *et al.*, 2017).

### CNN architectures

**Architecture of a convolutional neural network:** CNN architectures are mainly designed for the recognition of patterns in images as presented by Krizhevsky *et al.* (2012). On the other hand for the recognition of gestures from EMG signals, they present greater difficulty in not having defined parameters as dimension and object to be recognized but they require to work with much more complex patterns that make difficult to create an efficient architecture in comparison with the high efficiency that the CNN architectures oriented to images.

**Dataset:** The network to be implemented must be able to recognize the 4 gestures shown in Fig. 3 and 6, i.e., each one is a classification category in the network output. For this, the EMG signal is acquired for a time interval of 1 sec, sufficient time to contain the information necessary to test the performance of the CNN. The network enters a database preprocessed by feature extracting through the

wavelet packet transform and Wavelet Power Spectrum (WPS) is in charge of processing the information extracted from the WPT (Wavelet Packet Tree) with the objective of recreating the characteristics map, presented and processed as a matrix. In which it is clear to appreciate the information of the coefficients in each one of the frequency bands (Fig. 6), presenting variations of tone in the violet to blue scales. Where the violet presents greater activity in the band and the highest values in the coefficients and blue presents null activity. It is going to use for generates the feature map (Fig. 6) that will be entered into the network (input).

The WPD provides a time-frequency decomposition of a signal in the manner illustrated in (Fig. 6), the time-frequency plane with rectangles of different size means that energy components of the signal within different rectangles of specific time and frequency coordinates can be discerned. Accordingly, the rectangles are an indication of the optimal resolution achieved by the time-frequency. The WPD stipulate an octave-based decomposition of the frequency domain and gives good frequency resolution in the lower frequencies. Also, the WPD provides a useful time resolution in the higher frequencies that gets worse as it pass through to lower frequencies.

**Architecture implemented:** Three different architectures were implemented to test their performance in gesture recognition. It is important to clarify that all 3 architectures used the same training database.

**Architecture 1:** This architecture consists of 3 (Convolution/ReLU-Convolution/ ReLU-Max Pooling) packages for feature extraction with a padding of 2 in the first package and 1 in the others with the objective of not losing information of the corners and edges in the first convolutions. Finally, 3 fully-connected layers are used. Table 1 it is possible to better appreciate the construction of architecture where S is Stride and P is Padding.

Table 1: Architecture 1

| Types                | -----Kelme----- |         |       | #Filters |
|----------------------|-----------------|---------|-------|----------|
| Input                |                 | 8×31    |       | -        |
| Convolution/ReLU     | 5×5             | S = 1   | P = 2 | 18       |
| Convolution/ReLU     | 5×4             | S = 1   | P = 2 | 36       |
| Maxpooling           | 2               |         |       | -        |
| S = 2                | P = 0           | -       |       |          |
| Convolution/ReLU     | 3×5             | S = 1   | P = 1 | 72       |
| Convolution/ReLU     | 3×5             | S = 1   | P = 1 | 108      |
| Maxpooling           | 1×2             | S = 1×2 | P = 0 | -        |
| Convolution/ReLU     | 3×3             | S = 1   | P = 1 | 288      |
| Convolution/ReLU     | 3×3             | S = 1   | P = 1 | 576      |
| Maxpooling           | 2               | S = 2   | P = 0 | -        |
| Fully-connected/ReLU |                 | 1       |       | 2880     |
| DropouLayer          |                 | 0.5     |       | -        |
| Fully-connected/ReLU |                 | 1       |       | 2880     |
| DropouLayer          |                 | 0.5     |       | -        |
| Fully-connected      |                 | 1       |       | -        |
| Softmax              |                 | 4       |       | -        |

Table 2: Architecture 2

| Types                | -----Kelme----- |         |       | #Filters |
|----------------------|-----------------|---------|-------|----------|
| Input                |                 | 8×31    |       | -        |
| Convolution          | 5×5             | S=1     | P = 2 | 18       |
| Convolution          | 5×4             | S = 1   | P = 2 | 36       |
| Maxpooling           | 2               | S = 2   | P = 0 | -        |
| Convolution          | 3×5             | S = 1   | P = 1 | 72       |
| Convolution          | 3×5             | S = 1   | P = 1 | 108      |
| Maxpooling           | 1×2             | S = 1×2 | P = 0 | -        |
| Convolution          | 3×3             | S = 1   | P = 1 | 288      |
| Convolution          | 3×3             | S = 1   | P = 1 | 576      |
| Max pooling          | 2               | S = 2   | P = 0 | -        |
| Fully-connected/ReLU |                 | 1       |       | 2880     |
| DropouLayer          |                 | 0.5     |       | -        |
| Fully-connected      |                 | 1       |       | -        |
| Softmax              |                 | 4       |       | -        |

**Architecture 2:** This architecture consists of 3 (Convolution-Convolution-MaxPooling), packages with a padding of 2 in the first package and the others of 1, ending with 2 fully-connected layers. Table 2 it can be seen the construction of architecture in a better way.

**Architecture 3:** This architecture consists of 3 (Convolution-Convolution-MaxPooling), packages ending in three fully-connected layers, as set in Table 3.

In order to obtain the best training behavior of each of the implemented networks it is necessary to first understand the graph training accuracy presented in Fig. 7 which allows to determine in the first instants if the training can present overfitting. However, it is important to monitor their behavior during training, to discriminate if the network is learning over the epochs.

Secondly, the behavior associated with the loss training, presented in Fig. 8 should be analyzed. This allows to determine in its first data if the chosen learning rate is adequate for the training. This also allows validating if the network is actually learning or is experiencing a decrease in learning loss.

**Architecture training and validation:** Each of the proposed architectures was trained with a previously processed dataset composed of 180 samples per category

Table 3: Architecture 3

| Type                 | -----Kelme----- |         |       | # Filters |
|----------------------|-----------------|---------|-------|-----------|
| Input                |                 | 8×31    |       | -         |
| Convolution          | 5×5             | S = 1   | P = 2 | 18        |
| Convolution          | 5×4             | S = 1   | P = 2 | 36        |
| Maxpooling           | 2               | S = 2   | P = 0 | -         |
| Convolution          | 3×5             | S = 1   | P = 1 | 72        |
| Convolution          | 3×5             | S = 1   | P = 1 | 108       |
| Maxpooling           | 1×2             | S = 1×2 | P = 0 | -         |
| Convolution          | 3×3             | S = 1   | P = 1 | 288       |
| Convolution          | 3×3             | S = 1   | P = 1 | 576       |
| Maxpooling           | 2               | S = 2   | P = 0 | -         |
| Fully-connected/ReLU |                 | 1       |       | 2880      |
| Dropoulayer          |                 | 0.5     |       | -         |
| Fully-connected/ReLU |                 | 1       |       | 2880      |
| Dropoulayer          |                 | 0.5     |       | -         |
| Fully-connected      |                 | 1       |       | -         |
| Softmax              |                 | 4       |       | -         |

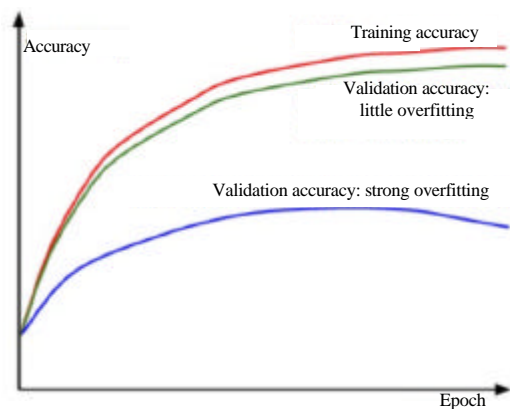


Fig. 7: Behavior of training accuracy (Johnson and Karpathy, 2017)

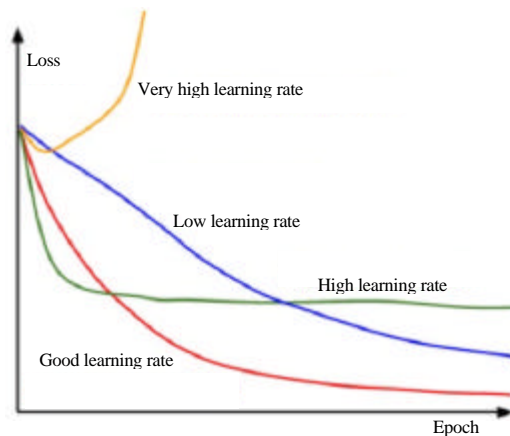


Fig. 8: Behavior of training loss (Johnson and Karpathy, 2017)

for a total of 720 training samples. According to the above it is important to note that the proposed architectures reached 100% training accuracy but the architecture that best behaves is architecture 3 as shown in Fig. 9, compared to the architectures 1 and 2 that present a

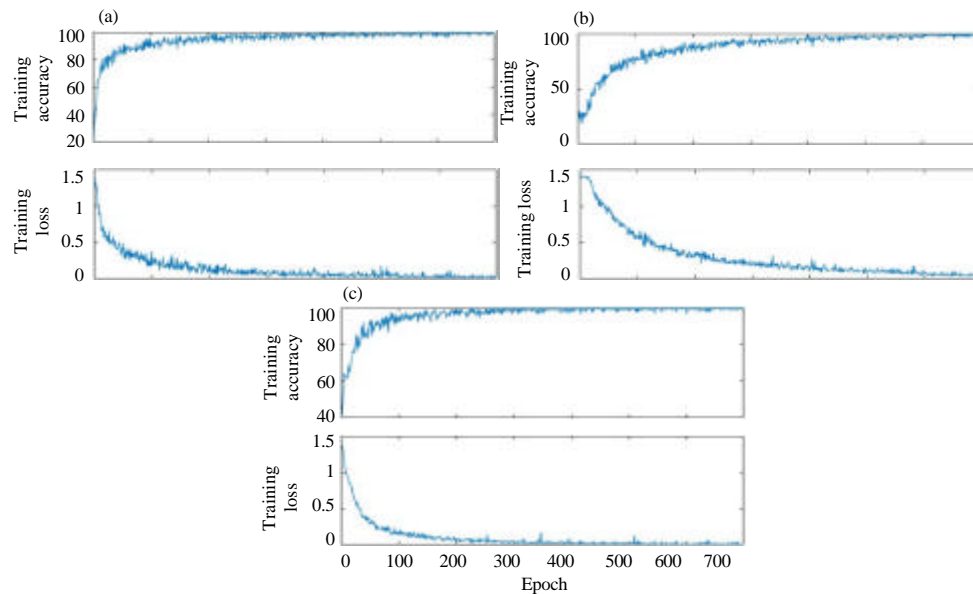


Fig. 9: Training accuracy and training loss: a) Architecture 1; b) Architecture 2 and c) Architecture 3

slower learning and with greater training loss. The main difference of the architectures lies in that the configuration of the architecture 3 does not use the Linear Rectification function (ReLU) among its convolution capable and in turn consists of three layers of fully-connected at the end.

Considering the above it will be used the successful architecture 3. Figure 9 shows the behavior of the training accuracy graph which begins to learn from the earliest epochs and reaches its first 100% accuracy around the 200 epoch where it reached its stabilization in the 450 epoch having a suitable behavior to discard overfitting. On the other hand, training loss shows that effectively the network is learning with the passing of the epochs and the behavior of the graph shows that the learning rate chosen is adequate.

Taking into account that it was obtained an accuracy of >95%, the validation of the network is performed, i.e., A database is made under the same conditions with which the dataset training base was acquired but in this case with a smaller amount and is assigned the name test, composed of 85 samples per category for a total of 340 samples.

The implemented validation technique corresponds to the confusion matrix Fig. 10 which is designed to evaluate the behavior of the network with a database different from that of training. This consists of rows corresponding to the predicted category (Output class) and columns that show the corresponding real category (Target class). The diagonal cells show the true positives of the category and finally, i.e., the correctly classified

| Confusion matrix |               |               |               |              |                |
|------------------|---------------|---------------|---------------|--------------|----------------|
| Output class     | 77<br>22.6%   | 2<br>0.6%     | 4<br>1.2%     | 0<br>0.0%    | 92.8%<br>7.2%  |
|                  | 0<br>0.0%     | 79<br>23.2%   | 3<br>0.9%     | 0<br>0.0%    | 96.3%<br>3.7%  |
|                  | 8<br>2.4%     | 3<br>0.9%     | 78<br>22.9%   | 0<br>0.0%    | 87.6%<br>12.4% |
|                  | 0<br>0.0%     | 1<br>0.3%     | 0<br>0.0%     | 85<br>25.0%  | 98.8%<br>1.2%  |
|                  | 90.6%<br>9.4% | 92.9%<br>7.1% | 91.8%<br>8.2% | 100%<br>0.0% | 93.8%<br>6.2%  |
| Target class     |               |               |               |              |                |

Fig. 10: Validation confusion matrix

members of the category in the lower right corner, the overall accuracy of the network in the recognition of the database is shown.

## RESULTS AND DISCUSSION

First, to evaluate the behavior of the feature extraction, Wavelet Discrete Transform (WDT) is used. Applying the wavelet packet transform tree which is presented in Fig. 11 where for each gesture two graphs are



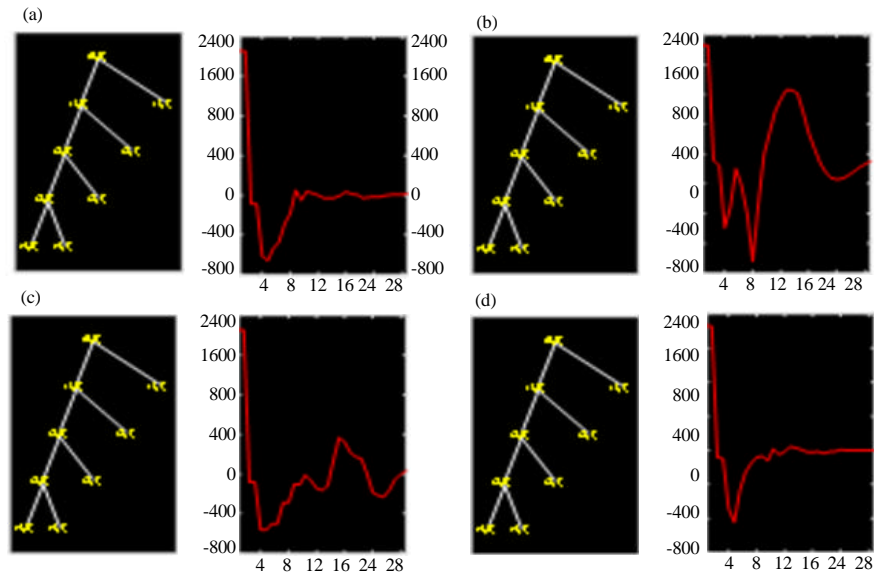


Fig. 11: Wavelet packet tree description: a) Close-fingers; b) Wave-in; c) Fist and d) Gun gesture

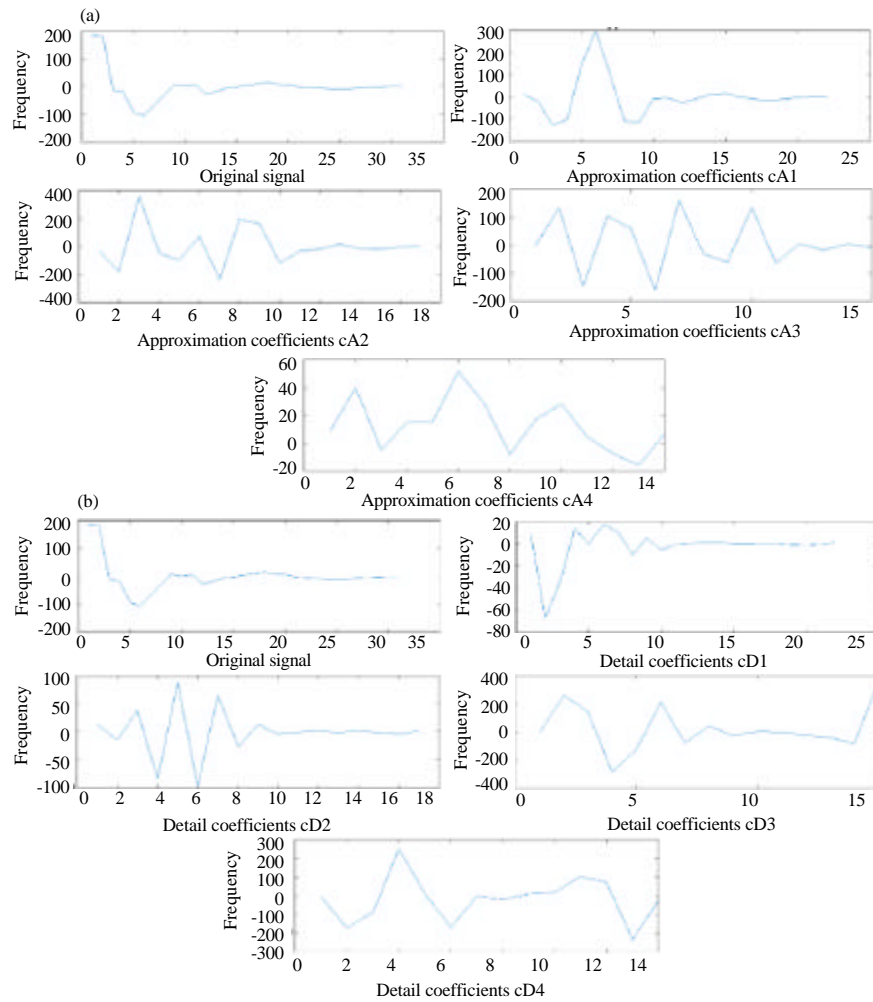


Fig. 12: Wavelet packet tree: a) Approximation coefficients and b) Detail coefficients

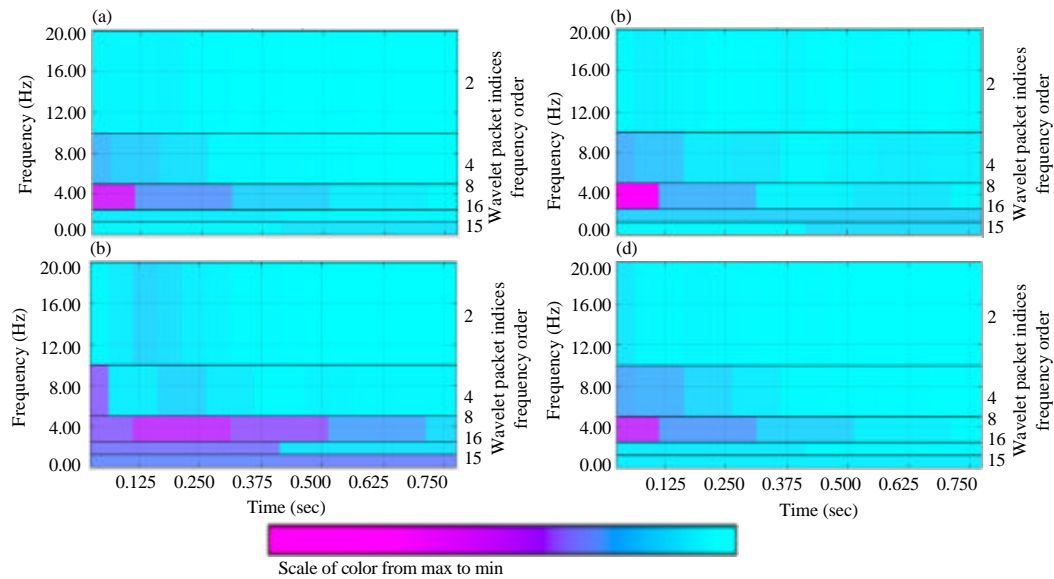


Fig. 13: Wavelet power spectrum behavior: a) Close-fingers; b) Fist; c) Wave-in and d) Gun

illustrated: to the left side, the frequency decomposition that makes the tree before the input signal and to the right side, the behavior of the signal applying the tree of frequency decomposition.

Next, Fig. 12 shows the behavior of the signal for each level of the wavelet packet tree. For example, to observe the behavior of the gesture close-fingers it is decomposed into approximation coefficients as shown in Fig. 12a and the detail coefficients, illustrated in Fig. 12b in charge of determining the frequencies in which the most important characteristics of the signal are found.

In the second place, after decomposing the signal and obtaining the approximation coefficients and the detailed coefficients, the packet spectrum is applied which allows to do the reconstruction of the signal from the coefficients and in this way, determine the frequency bands in which the greatest number of features are accumulated as can be shown in Fig. 13 which is characterized in the range of purple tones presenting a greater tone in the frequencies where there is greater activity. Also it fades its color in the frequencies with lower activities until presenting a blue tonality that presents inactivity.

Thirdly, the feature map is entered into the proposed network and the network validation is performed with the “test” database Fig. 14, obtaining a 96.5% of accuracy, from where it is possible to observe the percentage of accuracy in the recognition of each one of the categories. According to the above, the lowest accuracy with a 91.8% for the “Gun” gesture, may be caused by 2 important factors: first with the gesture “Fist” by the similarity of the

| Confusion matrix |               |               |               |              |               |
|------------------|---------------|---------------|---------------|--------------|---------------|
| Output class     | Close-fingers | Fist          | Gun           | Wave-in      |               |
| Close-fingers    | 82<br>24.1%   | 0<br>0.0%     | 3<br>0.9%     | 0<br>0.0%    | 96.5%<br>3.5% |
| Fist             | 0<br>0.0%     | 83<br>24.4%   | 4<br>1.7%     | 0<br>0.0%    | 95.4%<br>4.6% |
| Gun              | 3<br>0.9%     | 2<br>0.6%     | 78<br>22.9%   | 0<br>0.0%    | 94.0%<br>6.0% |
| Wave-in          | 0<br>0.0%     | 0<br>0.0%     | 0<br>0.0%     | 85<br>25.0%  | 100%<br>0.0%  |
|                  | 96.5%<br>3.5% | 97.6%<br>2.4% | 91.8%<br>8.2% | 100%<br>0.0% | 96.5%<br>3.5% |
|                  | 1             | 2             | 3             | 4            |               |
|                  | Close-fingers | Fist          | Gun           | Wave-in      |               |

Fig. 14: Validation of the architecture 3

gesture in which simply a movement variation is done in 2 fingers of the hand, this similarity is evidenced in the map of features in Fig. 13c, d. Second with the gesture “Close-fingers” does not present similarity in the gesture but there is activity of the same muscles with small variations of force when making each gesture, this similarity is evidenced in the feature map in Fig. 13a, b.

Fourth, the performance of the architecture is determined by evaluating the degree of membership with which the gesture was recognized, considering that the validation of the network is performed in real time by



Table 4: Gesture recognition results

| Degree of membership (0-100%) |                   |          |             |         |
|-------------------------------|-------------------|----------|-------------|---------|
| Tests                         | Close-fingers (%) | Fist (%) | Wave-in (%) | Gun (%) |
| 1                             | 99.988            | 100      | 100         | 99.844  |
| 2                             | 100               | 99.969   | 100         | 99.875  |
| 3                             | 99.998            | 99.994   | 100         | 91.495  |
| 4                             | 100               | 99.676   | 100         | 82.831  |
| 5                             | 100               | 99.799   | 100         | 54.190  |

means of 5 repetitions per gesture, thus obtaining the percentage of accuracy and the recognition label of the gesture performed (Table 4).

Finally, after the analysis of the three architectures that have been implemented it was obtained as a result that architecture 1, reached an accuracy of 93.8325% and architecture 3 with the same configuration unlike it was not built with ReLU activation functions between its convolution layers, reached the highest accuracy percentage with 96.4706% of accuracy.

On the other hand, the architecture 2 obtained an accuracy of 95.8824% while the architecture 3 which reached the highest percentage with a similar architecture with the difference that in the end consists of 3 fully-connected, allows to evaluate the importance of this last layer in the analysis of the characteristics obtained by the last layer of convolution, obtaining better performance applying 3 layers fully-connected increasing the percentage of effectiveness in a 0.6%.

## CONCLUSION

The results obtained through techniques based on deep learning for the recognition of patterns have revolutionized several fields of engineering. For this reason, this technique is applied to the processing of signals, more specifically in EMG signals, allowing the development of much more efficient and functional devices. For example, contribute to the development of protein devices or robots focused on tele-operation.

The 3 architectures proposed are novel, since, they were not implemented in previous studies, satisfactory results were obtained for the 3 architectures reaching a percentage of accuracy >90%. However, it is important to emphasize the technique of feature extraction implemented, since, this technique allows to evaluate in a more detailed way the behavior of the signal in different frequency bands, managing to capture to the minimum details that allow to differentiate a signal from the other.

On the other hand, the implementation of the 3 architectures has allowed to evaluate 2 important points: First, the function ReLU, although, theoretically converts the negative values to 0. In this case, for signals

where its feature map does not contain negative values but if very close to 0 it would be recognizing them as 0, generating a significant loss of the information evidenced in the efficiency of the network. Second, the importance of fully-connected in this type of architecture, since, the similarity of signal behavior is high and success is in the greater recognition of characteristics that differentiate one signal from the other. It was necessary to implement 2 fully-connected specifically to learn a greater number of characteristics and a third one for connection with the softmax and final classification.

## IMPLEMENTATIONS

Considering the results obtained, convolutional neural network architectures for the recognition of gestures can be implemented in applications such as limb rehabilitation, neuromuscular disease recognition, protein device construction or man-machine interaction for the control of a teleoperated robot.

## ACKNOWLEDGEMENT

The researchers are grateful to the Nueva Granada Military University which through, its Vice chancellor for research, Finances the present project with code IMP-ING-2290 and titled "Prototype of robot assistance for surgery", from which the present study is derived.

## REFERENCES

- Anonymous, 2017. MySignals HWv2-eHealth and medical IoT development platform for arduino. Libelium Comunicaciones Distribuidas S.L., Zaragoza, Spain. <https://www.cooking-hacks.com/mysignals-hw-ehealth-medical-biometric-iot-platfor-m-arduino-tutorial/>
- Atzori, M., M. Cognolato and H. Muller, 2016. Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands. *Front. Neurorob.*, 10: 1-10.
- Cardinali, D.P., 2007. [Applied Neuroscience: Its Basics]. Editorial Medica Panamericana S.A., Bogota, Colombia, ISBN:978-950-06-0328-7, Pages: 505 (In Spanish).
- Cipriani, C., F. Zaccane, S. Micera and M.C. Carrozza, 2008. On the shared control of an EMG-controlled prosthetic hand: Analysis of user-prosthesis interaction. *IEEE. Trans. Rob.*, 24: 170-184.

- Daubechies, I., 1992. Ten Lectures on Wavelets. Vol. 61, Society for Industrial Applied Maths, Philadelphia, pp: 357.
- Englehart, K., B. Hudgin and P.A. Parker, 2001. A wavelet-based continuous classification scheme for multifunction myoelectric control. IEEE. Trans. Biomed. Eng., 48: 302-311.
- Fukuda, O., T. Tsuji, M. Kaneko and A. Otsuka, 2003. A human-assisting manipulator teleoperated by EMG signals and arm motions. IEEE. Trans. Rob. Autom., 19: 210-222.
- Johnson, J. and A. Karpathy, 2017. CS231n convolutional neural networks for visual recognition. Stanford University Computer Science, Stanford, California, USA.
- Krizhevsky, A., I. Sutskever and G.E. Hinton, 2012. Image net classification with deep convolutional neural networks. Proc. Neural Inf. Process. Syst., 1: 1097-1105.
- LeCun, Y., Y. Bengio and G. Hinton, 2015. Deep learning. Nat., 521: 436-444.
- Misiti, M., Y. Misiti, G. Oppenheim and J.M. Poggi, 2017. Wavelet Toolbox™ User's Guide. MathWorks Publisher, Natick, Massachusetts, USA.,.
- Park, K.H. and S.W. Lee, 2016. Movement intention decoding based on deep learning for multiuser myoelectric interfaces. Proceedings of the 4th International Winter Conference on Brain-Computer Interface (BCI'16), February 22-24, 2016, IEEE, Yongpyong, South Korea, ISBN:978-1-4673-7841-3, pp: 1-2.
- Phinyomark, A., P. Phukpattaranont and C. Limsakul, 2011. Wavelet-based denoising algorithm for robust EMG pattern recognition. Fluctuation Noise Lett., 10: 157-167.
- Romo, H., J. Realpe and P. Jojoa, 2007. Hand gesture recognition using kinect. Rev. Av. Sistemas Inf., 4: 127-136.
- Serre, T., G. Kreiman, M. Kouh, C. Cadieu and U. Knoblich *et al.*, 2007. A quantitative theory of immediate visual recognition. Prog. Brain Res., 165: 33-56.
- Shinde, A. and Z. Hou, 2005. A wavelet packet based sifting process and its application for structural health monitoring. Struct. Health Monit., 4: 153-170.
- Wu, J.D. and C.H. Liu, 2008. Investigation of engine fault diagnosis using discrete wavelet transform and neural network. Expert Syst. Appl., 35: 1200-1213.