Real-Time Hand Gesture Recognition Model Using Deep Learning Techniques and EMG Signals

Edison A. Chung
Department of Informatics and Computer Science
Escuela Politécnica Nacional
Quito, Ecuador
edison.chung@epn.edu.ec

Marco E. Benalcázar
Department of Informatics and Computer Science
Escuela Politécnica Nacional
Quito, Ecuador
marco.benalcazar@epn.edu.ec

Abstract— **Gesture recognition has multiple applications in medicine, engineering and robotics. It also allows us to develop new and more natural approaches to human-machine interaction. Real-time hand gesture recognition consists of identifying, with no perceivable delay, a given gesture performed by the hand at any moment. In this paper, we propose a model for real-time hand gesture recognition. The proposed model takes as input electromyographic (EMG) signals measured on the forearm, using the commercial sensor Myo Armband. We use an autoencoder for automatic feature extraction, and an artificial feed-forward neural network for classification. The proposed model can recognize the same 5 gestures as the proprietary recognition system of the Myo Armband, achieving an average recognition accuracy of 85.08% ± 15.21%, with an average response time of 3 ± 1 ms. The proposed model is general, which implies that it can recognize the gestures from any user, even when his\her data are not included in the training dataset. Finally, for reproducing this work, we make publicly available the code that implements the proposed model.**

Keywords— hand gesture recognition; automatic feature extraction; artificial neural networks; autoencoders

I. Introduction

Hand gesture recognition is the problem of identifying the class, from a predefined set of classes, and the instant of occurrence of a given movement of the hand [1]. Hand gesture recognition allows us the development of new and more natural approaches to human-machine interaction. The gesture recognition models are focused on the human side of the interaction. Solving the problem of gesture recognition involves several challenges. For example, two different people can perform the same gesture in a different way. The data required for gesture recognition can be acquired through several types of sensors: gloves, cameras and ultrasound sensors, each one with its own challenges to overcome. Cameras are very sensible to variations of light intensity and have problems with occlusion and changes of the distance between the hand and the camera [2]. Gloves might not have the right size for the hands of a user and thus they can be uncomfortable for some people depending on the application and time of use [1]. With ultrasound sensors a cross-talk effect can happen when a sensor receives the waves emitted from a previous emission, or from dispersion.

For this work, we use electromyographic sensors to capture the electric signals produced by the muscular activity in the forearm. These electric signals are called electromyographic signals, or simply electromyography (EMG) [3] from now on. The classification of these signals can be used in multiple domains of application, including: videogame interfaces, robotics, sign language translation to text or voice, and bionics [4, 5, 6]. EMG is a noisy signal that can vary in amplitude and frequency due to electromagnetic induction or even due to interaction with electrical signals from different tissues of the human body. However, EMG signals carry the information of the commands given by the human brain to contract a skeletal muscle to produce force and/or movement and, thus, EMGs are a direct representation of a user's movement intention [7]. For this reason, EMG signals are a good candidate to be the input signal of a hand gesture recognition system.

Several hand gesture recognition models have been proposed for multiple purposes. For example, in [8], using a Support Vector Machine (SVM) classifier and EMG signals, the proposed low consumption system achieves an 88% accuracy for 3 gestures with only five training repetitions. In [9], the proposed system can recognize up to 4 gestures with a recognition accuracy of 87%, using a set of electrodes, an SVM classifier, and ten training repetitions. In [10], the proposed system is used to drive a remote-controlled car using a single EMG electrode and a set of 4 gestures, with an accuracy of 94% by combining two simple linear classifiers.

Artificial neural networks (ANNs) have also been proposed for implementing hand gesture recognition models. For example, in [11], the proposed system achieved 100% of accuracy on 4 gestures, using a set of EMG electrodes and a Multilayer Perceptron with only one hidden layer. In [12], the proposed system uses a feed-forward ANN to classify the signals from 4 gestures, using 2 EMG electrodes, with an accuracy of 83.5%. In [13], the proposed system is used to control a quad copter using 4 gestures, a sensor with 4 EMG electrodes, and a feed-forward ANN, obtaining a recognition accuracy of 93%; however, this system requires 20 training repetitions per gesture. In [14], the proposed system can recognize 3 gestures with a single channel EMG signal processed using the wavelet transform and a feed-forward ANN, obtaining an average recognition accuracy of 93.25%. In [15], a feed-forward ANN is used to recognize 6 gestures with a 3 channel EMG signal, obtaining a recognition accuracy of 71%. As we can see above, most of the works proposed in the scientific literature have a high recognition accuracy for a low number of gestures. However, these works propose user-specific models that require training for each person and for each time they are used. This requirement of frequent training of user-specific models is a big limitation for their use in practical applications, which usually require plug-and-play systems. Additionally, the estimation of the recognition accuracy might be too optimistic since the low number of testing examples. For these reasons, in this work, we propose a real-time hand gesture recognition model for daily use that is general and has a recognition accuracy, tested on 60 users different from

978-9-0827-9703-9/19/$31.00 ©2019 IEEE

the training users, that is competitive with the models proposed in the scientific literature.

The hand gesture recognition model proposed in this work uses the Myo Armband to collect the EMG signals on the forearm of a user. We implemented a feed-forward multilayer neural network to classify the EMG signals. We use this machine learning model since this type of neural network has the potential of being a universal learning machine [16]. In other words, this network architecture has the potential of implementing any decision border. The architecture of the proposed network consists of 4 layers: an input layer, two hidden layers and an output layer. The input layer takes the rectified raw EMG. The first hidden layer is an autoencoder used for automatic feature extraction. The second hidden layer, along with the output layer, implement the classifier.

The proposed model can process and classify EMG signals in less than 300ms [17] (i.e., processing in real time). It recognizes, with higher accuracy, the same 5 gestures as the proprietary software of the Myo Armband. The gestures recognized in this work are shown in Fig. 1. The proposed model works for any user and it does not require any training or any type of tuning before its use, since it was already trained as a general recognition model.

Fig. 1. Gestures recognized by the proposed model.

II. Materials

In this section, we describe the characteristics of the Myo Armband, and the nature of the gesture samples recorded in the database that we used.

A. Myo Armband

The Myo Armband, shown in Fig. 2, is a gesture and motion control device manufactured by Thalmic Labs. It weighs approximately 93g and consists of 8 EMG sensors, which operate at a sampling frequency of 200Hz. This device also has an Inertial Measurement Unit (IMU), which works at a sampling frequency of 50Hz. In this work, however, we only use the EMG data. The Myo Armband uses Bluetooth to transmit its data to the computer. It also comes with a proprietary software that recognizes the 5 hand gestures shown in Fig. 1.

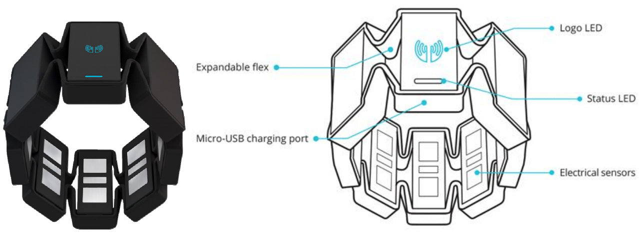

Fig. 2. Myo armband by Thalmic Labs.

B. Database of gestures

We used the Myo Armband to record the EMG signals of the 5 gestures shown in Fig. 1. These gestures were performed by 120 different users. We recorded 50 samples of each gesture per user, with each sample being contained in a 5 second window. These 50 samples were split into 2 subsets of 25 samples each, one for training and the other for testing. We also recorded 10 samples of the arm in a relaxed or rest position. In total, we recorded 37200 samples.

All users were asked to wear the Myo Armband on the right forearm, with its LED indicator on top aimed towards their hand and with their palm facing to the ground, as shown in Fig. 3. For each sample, the user was asked to start in the relax or rest position, perform the gesture only once when signaled by the recording program, and then return to the relax position, all of these within a 5 second window.

Fig. 3. Location of the Myo armband on the forearm.

We saved basic information about the user including gender (75% males, 25% females), age (17-29 years), dominant hand (9% left handed, 81% right handed), and whether or not they had any injury in their right arm (16% had suffered an injury). The data recorded includes the EMG for each gesture, the name of each gesture, and the response from the proprietary software of the Myo Armband.

The recorded EMG data for a single sample of any gesture consist of 8 channels, one for each sensor of the Myo Armband, with their values within the interval [-1, 1]. At a sampling rate of 200Hz over 5 seconds, we have 1000 values per channel. However, the actual time length in which the gesture was performed is shorter than 5 seconds.

We used the data from 60 out of the 120 users for developing the proposed model. These 60 users were selected randomly. From these 60 users, 50 were used to train the model and 10 were set aside for model selection. The data that we used to train the proposed model correspond to all the 50 EMG samples from the recorded gestures of these 50 users with their respective labels. The data corresponding to the testing EMG samples from the remaining 60 users were used for the final evaluation of the fully trained model.

III. Methodology

To develop the model proposed in this work, we followed a methodology composed of the following steps.

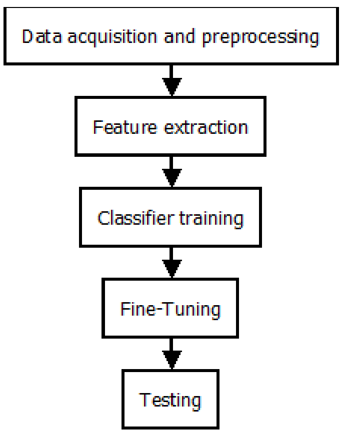

Fig. 4. Workflow for training the hand gesture recognition model.

A. *Training the Recognition Model*

The process for training the recognition model consists of 5 steps, which are shown in Fig. 4. The architecture of the ANN of the proposed model is shown in Fig. 5.

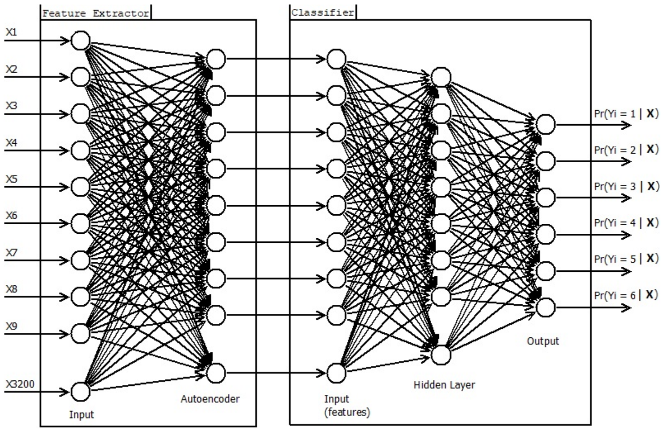

Fig. 5. Architecture of the ANN for the proposed model.

1) Data acquisition and preprocessing: The preprocessing for each sample begins with the rectification of the signal by computing the absolute value of all the values of its channels. Then, we use the muscle activity detector proposed in [1] to obtain the segment of the EMG in which the gesture was performed within the sample. In other words, we use the detector to remove the head and tail of each sample, both of which contain no useful information about the gesture. This occurs because the samples of each gesture start and end with the arm in the relax position. If no muscle activity is detected in the sample (e.g., in relax samples), we use all the points of the sample.

Afterwards, to simulate how a real time system would work in practice, we use a sliding window approach, obtaining several window observations. We found that the best recognition accuracy results were obtained by using a window of 400 points, with a stride of 10 points. If the size of the segmented EMG is smaller than the size of the sliding window, then zeros are added evenly before and after the segmented EMG to complete the window length. Each window observation obtained is transformed into a single vector by concatenating the values from its 8 channels, starting with the values from the first channel, then the values from the second channel and so on. All the vectors obtained in this way, from all the samples, are put together in a single matrix. This matrix contains 779653 rows and 3200 columns. Each row corresponds to a single window observation. Then, this matrix is divided by subsampling randomly its rows, where 50% of the observations become an unsupervised training set, 25% become a supervised training set, and the remaining 25% become a validation set.

2) Feature extraction: In this step, we trained an autoencoder using the unsupervised training set described previously. This autoencoder will be used for automatic feature extraction. An autoencoder is a neural network designed with the objective of learning the identity function, so that its input is (approximately) equal to its output. By using less neurons in the hidden layer than the number of outputs, the autoencoder must learn a compressed representation of the input data, in other words, it encodes the input data [18].

To train the autoencoder, we used the L-BFGS method to optimize our cost function, which is the mean squared error between the input and the output vectors of the autoencoder. The trained autoencoder is then used to encode the samples from the supervised training set, resulting in a set of feature vectors which will be used to train the classifier, along with their corresponding labels. The size of each vector is equal to the size of the hidden layer of the autoencoder.

After testing several configurations including 1, 2 and 3 stacked autoencoders, with varying results, we determined that the overall best configuration for our work was to use a single autoencoder, with a size of 200 neurons in the hidden layer, a weight decay parameter of 3.0×10^{-3}, a weight of the sparsity penalty term of 0.5, and a sparsity proportion of 0.1. We used the log-sigmoid transfer function.

3) Classifier training: We trained a classifier using the feature vectors obtained by passing the supervised training set through the autoencoder whose design was described in the previous step. The classifier that we used is based on a feed-forward ANN with 3 layers: the input layer takes feature vectors, and the hidden layer along with the output layer implement the classification module, which returns a column vector with 6 rows containing the probabilities of the input belonging to each class. Using a value of 0.8 as the conditional probability threshold, we take the class with the highest probability and assign it to the respective sample. If all probabilities are below the threshold, we label the input with the class 1 (i.e., no gesture or relax).

The transfer functions that we tested include the rectified linear unit (ReLU), tanh, log-sigmoid, softplus, and the exponential linear unit (ELU). From all these functions, the best results were obtained using a hidden layer of 150 neurons with the ReLU function, and a regularization factor for weight penalty of 0.1 to train the ANN.

4) Fine-Tuning: The final architecture of the whole model corresponds to a feed-forward ANN with 4 layers: input, autoencoder, ReLU hidden layer, and output. We fine-tuned this ANN using low (1 to 10) and high (250 to 500) numbers of iterations, with the inputs and labels from the unsupervised training set only, with the data from the supervised training set only, and with the data from both sets combined. The best results were obtained using 500 iterations and the data from the supervised training set. This process increased the average recognition accuracy of the model on the training dataset by a significant margin, from 83.28% to 87.44%.

5) Testing: In this step, we estimate the recognition accuracy of the proposed model. For this task, we use the data of the 60 users of the testing dataset, which was not used either for training nor for model selection. In total 7500 samples were used for testing, with 1500 samples per gesture and a total of 5 gestures. For testing, each sample is preprocessed in the same manner as described in the first step of this section, except this time the muscle activity detector is not used to speed up the classification process.

For evaluating the recognition accuracy, we obtained a single label that describes the gesture contained in the testing sample. Then, we compared the estimated label with the actual label of the sample. If both labels match, the recognition is considered successful. Otherwise, there is an error in the recognition. The output of the classifier applied to each testing sample is a vector of labels, denoted as R, with each element being the class of the classification of one window observation. Ideally, R will begin with labels corresponding to class 1 (i.e., relax position or no gesture), then have some labels corresponding to the recognized gesture, and end with labels corresponding to class 1 again. However, in practice, R can have outliers which are labels from other gestures because of erroneous classifications. To remove these outliers, each label $C_i \in R$, with $i > 1$, is compared with the previous label C_{i-1}. If C_i is different from C_{i-1}, then C_i is changed to the label of the relax position as can be seen in Fig. 6.

1	1	1	3	3	4	3	3	3	1	1	1	1	1	1

A vector R from a single sample.

1	1	1	1	3	1	1	3	3	1	1	1	1	1	1

After removing outliers.

Fig. 6. Removing outliers from the vector of labels R.

To obtain a single label for the whole testing sample, we first removed the labels corresponding to the class no-gesture from R. Then, we defined 3 different approaches to obtain the class of the testing sample: the first approach is to label R with the most frequent label that appears in it, which will be denoted as "Mode". The second approach is to label R with the first gesture classification that appears in it, which will be denoted as "First Transition". The third approach is to obtain a vector U of sorted unique elements in R and then take the first gesture in U as the class of that sample, which will be denoted as "Unique."

IV. Results and Discussion

A. Recognition Accuracy

For the 60 users of the testing set, we obtained an average recognition accuracy of 85.08 ± 15.21% for the Mode approach, 82.29 ± 14.71% for the First Transition approach, and 82.32 ± 15.43% for the Unique approach. We present the confusion matrix for each approach in Tables I, II, and III, respectively. For the Mode approach, the highest and the lowest sensitivities occur for the gestures wave left (87.3%) and fingers spread (78.8%), respectively. The highest and the lowest precisions occur for the same gestures, with 93.1% for wave left and 82.5% for fingers spread. For the First Transition approach, the highest and the lowest sensitivities occur for the gestures wave left (85.1%) and fingers spread (77.3%), respectively. The highest and the lowest precisions occur for the same gestures, with 89.9% for wave left and 83.2% for fingers spread. For the Unique approach, the highest and the lowest sensitivities occur for the gestures fist (88.8%) and fingers spread (75.2%), respectively. The highest and the lowest precisions occur for the gestures wave left (90.2%) and fist (81.7%), respectively. Upon further inspection, we can see that the proposed model, with the 3 approaches, fails the most at discriminating between the fingers spread and wave right. With the Mode approach, the model will not recognize any gesture at all in 2.1% of the cases. With both the First Transition and the Unique approaches, the model will not recognize any gesture at all in 4.2% of the cases. The best accuracy of this work (85.08%) is higher than the accuracy of the Myo Armband (83.1%), but lower than the accuracies obtained in previous works 86.0% [19] and 89.5% [1]. However, we must consider that the model proposed in this paper is general unlike the models of [19] and [1], which are user-specific.

The high standard deviations of the recognition accuracies indicate that the proposed model has a performance that varies significantly from user to user. This may occur because of the differences in how each person wears the sensor and makes the gestures (i.e., speed and force movements). Additionally, the EMG is a non-stationary process that varies from person to person; and for a given gesture and a given person, the EMG changes from one time to another. These are some reasons why gesture recognition using EMG is a very challenging problem.

B. Real-time operation

To evaluate the real-time operation of the proposed model, we implemented a simple program that uses a pretrained neural network architecture and connects to the Myo Armband. This program shows the EMG signals being processed and the gesture that was recognized by the model.

The tests of the processing time of the proposed model were run on a desktop computer with an Intel® Core™ i7-3770S processor and 8GB of RAM. From these tests, we determined that the time required to process and classify a single window observation is, in average, 3 ± 1ms. Theoretically, with a length of the stride of 10 points at 200Hz, the total time between each response is 50ms, from which the system only requires, on average, 3ms to classify each window observation.

TABLE I. Confusion Matrix For The Mode Approach

Predictions	Targets: FIST	WAVE LEFT	WAVE RIGHT	FINGERS SPREAD	DOUBLE TAP	% PRECISION % ERROR
NO GESTURE	**0** 0.0%	**35** 0.5%	**8** 0.1%	**82** 1.1%	**29** 0.4%	**0.0%** 100%
FIST	**1296** 17.3%	**81** 1.1%	**1** 0.0%	**43** 0.6%	**66** 0.9%	**87.2%** 12.8%
WAVE LEFT	**73** 1.0%	**1310** 17.5%	**3** 0.0%	**3** 0.0%	**18** 0.2%	**93.1%** 6.9%
WAVE RIGHT	**22** 0.3%	**68** 0.9%	**1287** 17.2%	**112** 1.5%	**17** 0.2%	**85.5%** 14.5%
FINGERS SPREAD	**17** 0.2%	**3** 0.0%	**167** 2.2%	**1182** 15.8%	**64** 0.9%	**82.5%** 17.5%
DOUBLE TAP	**57** 0.8%	**30** 0.4%	**31** 0.4%	**78** 1.0%	**1306** 17.4%	**87.0%** 13.0%
%SENSITIVITY % ERROR	**86.4%** 13.6%	**87.3%** 12.7%	**85.8%** 14.2%	**78.8%** 21.2%	**87.1%** 12.9%	**85.08%** 14.92%

TABLE II. Confusion Matrix For The First Transition Approach

Predictions	Targets: FIST	WAVE LEFT	WAVE RIGHT	FINGERS SPREAD	DOUBLE TAP	% PRECISION % ERROR
NO GESTURE	**50** 0.7%	**22** 0.3%	**46** 0.6%	**135** 1.8%	**59** 0.8%	**0.0%** 100%
FIST	**1226** 16.3%	**92** 1.2%	**8** 0.1%	**42** 0.6%	**60** 0.8%	**85.9%** 14.1%
WAVE LEFT	**96** 1.3%	**1277** 17.0%	**18** 0.2%	**2** 0.0%	**28** 0.4%	**89.9%** 10.1%
WAVE RIGHT	**32** 0.4%	**60** 0.8%	**1255** 16.7%	**98** 1.3%	**30** 0.4%	**85.1%** 14.9%
FINGERS SPREAD	**25** 0.3%	**5** 0.1%	**136** 1.8%	**1160** 15.5%	**69** 0.9%	**83.2%** 16.8%
DOUBLE TAP	**71** 0.9%	**44** 0.6%	**37** 0.5%	**63** 0.8%	**1254** 16.7%	**85.4%** 14.6%
%SENSITIVITY % ERROR	**81.7%** 18.3%	**85.1%** 14.9%	**83.7%** 16.3%	**77.3%** 22.7%	**83.6%** 16.4%	**82.29%** 17.71%

TABLE III. Confusion Matrix For The Unique Approach

Predictions	Targets: FIST	WAVE LEFT	WAVE RIGHT	FINGERS SPREAD	DOUBLE TAP	% PRECISION % ERROR
NO GESTURE	50 0.7%	22 0.3%	46 0.6%	135 1.8%	59 0.8%	0.0% 100%
FIST	**1332** 17.8%	**140** 1.9%	**9** 0.1%	**65** 0.9%	**85** 1.1%	**81.7%** 18.3%
WAVE LEFT	**62** 0.8%	**1255** 16.7%	**35** 0.5%	**4** 0.1%	**36** 0.5%	**90.2%** 9.8%
WAVE RIGHT	**14** 0.2%	**59** 0.8%	**1261** 16.8%	**110** 1.5%	**38** 0.5%	**85.1%** 14.9%
FINGERS SPREAD	**8** 0.1%	**3** 0.0%	**123** 1.6%	**1128** 15.0%	**84** 1.1%	**83.8%** 16.2%
DOUBLE TAP	**34** 0.5%	**21** 0.3%	**26** 0.3%	**58** 0.8%	**1198** 16.0%	**89.6%** 10.4%
%SENSITIVITY % ERROR	**88.8%** 11.2%	**83.7%** 16.3%	**84.1%** 15.9%	**75.2%** 24.8%	**79.9%** 20.1%	**82.32%** 17.68%

However, there is a slight delay between the moment in which the user starts the gesture and the moment in which the system recognizes that the user is performing a gesture. This is due to the use of a sliding window. There must be a certain amount of overlap between the sliding window and the signal from the gesture for the system to accurately recognize the gesture. This means that in the worst case, the user will have to complete the gesture before the system recognizes it.

Finally, to reproduce this work we make publicly available the code that implements the proposed model at https://drive.google.com/drive/folders/1eODcbPTAs-53Qr7YcOhu11IWzJjH_eRh?usp=sharing .

V. Concluding Remarks

In this paper, we have presented a real-time hand gesture recognition model based on deep learning techniques and EMG signals acquired with the Myo Armband. The deep learning techniques are used for automatic feature extraction and classification. The proposed model can recognize 5 different hand gestures: fist, wave left, wave right, fingers spread and double tap; with an average recognition accuracy of 85.08 ± 15.21%. The time of processing of the proposed model is 3 ± 1 ms.

Since the proposed model is general, it does not need the user to train it beforehand and can be used by anyone. However, the model can only be used with the Myo Armband worn on the right forearm and in the correct position (Section II-B). Depending on the user, the model may have varying results, evidenced by the high standard deviation of the recognition accuracy.

For future work, recurrent neural networks should be considered for the recognition, since they can make use of their internal state to process a sequence of inputs. Their internal state would allow them to use previous recognitions to predict the current one with greater accuracy. Additionally, a method for making the proposed general model robust to rotations of the sensor should be developed.

Acknowledgment

The authors gratefully acknowledge the financial support provided by Escuela Politécnica Nacional for the development of the research project PIJ-16-13.

References

[1] M. E. Benalcázar *et al.*, "Real-time hand gesture recognition using the Myo armband and muscle activity detection," *2017 IEEE Second Ecuador Technical Chapters Meeting (ETCM)*, Salinas, 2017, pp. 1-6.

[2] G. Marin, F. Dominio and P. Zanuttigh, "Hand gesture recognition with leap motion and kinect devices," *2014 IEEE International Conference on Image Processing (ICIP)*, Paris, 2014, pp. 1565-1569.

[3] A. Ganiev, H. Shin and K. Lee., "Study on Virtual Control of a Robotic Arm via a Myo Armband for the Self-Manipulation of a Hand Amputee," *International Journal of Applied Engineering Research*, 2016, vol. 11, no 2, pp. 775-782.

[4] G. Luh, H. Lin, Y. Ma and C. J. Yen, "Intuitive muscle-gesture based robot navigation control using wearable gesture armband," *2015 International Conference on Machine Learning and Cybernetics (ICMLC)*, Guangzhou, 2015, pp. 389-395.

[5] A. Norali, M. Som and J. Kangar-Arau, "Surface electromyography signal processing and application: A review," *Proceedings of International Conference on Man-Machine Systems (ICoMMS),* 2009, pp. 11-13.

[6] Manuel A León *et al.*, "Virtual Rehabilitation System for Fine Motor Skills Using a Functional Hand Orthosis," *Proceedings of the International Conference on Augmented Reality*, 2018, pp. 78-94.

[7] L. Tagliapietra, M. Vivian, M. Sartori, D. Farina and M. Reggiani, "Estimating EMG signals to drive neuromusculoskeletal models in cyclic rehabilitation movements," *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Milan, 2015, pp. 3611-3614.

[8] S. Benatti *et al.*, "A sub-10mW real-time implementation for EMG hand gesture recognition based on a multi-core biomedical SoC," *2017 7th IEEE International Workshop on Advances in Sensors and Interfaces (IWASI)*, Vieste, 2017, pp. 139-144.

[9] Z. Wu and X. Li, "A wireless surface EMG acquisition and gesture recognition system," *2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, Datong, 2016, pp. 1675-1679.

[10] J. Kim, S. Mastnik, and E. André, "EMG-based Hand Gesture Recognition for Realtime Biosignal Interfacing," *13th International Conference on Intelligent User Interfaces - IUI '08*, Canary Islands, 2008, pp. 30-39.

[11] V. T. Gaikwad and M. M. Sardeshmukh, "Sign language recognition based on electromyography (EMG) signal using artificial neural network (ANN)," *International Journal of Industrial Electronics and Electrical Engineering*, 2014, vol. 2, no 6, pp. 73-75.

[12] E. H. Shroffe and P. Manimegalai, "Hand Gesture Recognition based on EMG Signals using ANN," *International Journal of Computer Application*, 2013, vol. 2, no 3, pp. 31-39.

[13] H. Li, X. Chen and P. Li, "Human-computer interaction system design based on surface EMG signals," *Proceedings of 2014 International Conference on Modelling, Identification & Control*, Melbourne, VIC, 2014, pp. 94-98.

[14] S. M. Mane, R. A. Kambli, F. S. Kazi and N. M. Singh, "Hand Motion Recognition from Single Channel Surface EMG Using Wavelet & Artificial Neural Network," *Proceedings of 4th International Conference on Advances in Computing, Communication and Control (ICAC3'15)*, 2015, vol. 49, pp. 58-65.

[15] J. Wang, H. Ren, W. Chen and P. Zhang, "A portable artificial robotic hand controlled by EMG signal using ANN classifier," *2015 IEEE International Conference on Information and Automation*, Lijiang, 2015, pp. 2709-2714.

[16] K. Hornik, "Approximation Capabilities of Multilayer Feedforward Networks," *Neural Networks*, 1991, vol. 4, pp. 251-257.

[17] H. Mizuno, N. Tsujiuchi and T. Koizumi, "Forearm motion discrimination technique using real-time EMG signals," *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Boston, MA, 2011, pp. 4435-4438.

[18] A. Ng, J. Ngiam, C. Foo, Y. Mai and C. Suen, "Autoencoders and Sparsity - Ufldl", *ufldl.stanford.edu*, para. 1-2, apr. 7, 2013. [Online]. Available: http://ufldl.stanford.edu/wiki/index.php/Autoencoders_and_Sparsity. [Accessed: aug. 11, 2018].

[19] M. E. Benalcázar *et al.*, "Hand gesture recognition using machine learning and the Myo armband," *Proceedings of the 25th European Signal Processing Conference (EUSIPCO)*, 2017, pp. 1040-1044.