2020 17th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)
Mexico City, Mexico. November 11-13, 2020

Comparison of EMG signal classification algorithms for the control of an upper limb prosthesis prototype

Daniel Lorias Espinoza
Electrical Departament Bioelectronic Section
CINVESTAV IPN
México City
dlorias@cinvestav.mx

Leobardo Elí Sánchez Velasco
Electrical Departament Bioelectronic Section
CINVESTAV IPN
México City
leli.sanchezv@cinvestav.mx

Abstract— in this paper, Machine Learning algorithms (Decision Trees and Support Vector Machines) are proposed and compared to select a classification system for EMG signals to improve the performance of pattern recognition for the control of a prosthesis prototype.

The training, validation and signal classification were made using the Classification Learner application of the MatLab software, using a database captured with the commercial myoelectric armband MYO which contain the information of eight different hand movements.

The results show that Support Vector Machines algorithms have a better performance than the decision trees, reaching the 99.8% of accuracy with linear and quadratic kernel and the 99.9% using a cubic kernel.

Keywords— classification, machine learning, EMG, Myo.

I. INTRODUCTION

Electromyographic signals have been used in many fields of biomedical engineering, such as teleoperation of devices [1], exoskeletons and rehabilitation devices [2], functional electrostimulation [3], prostheses [4], among others.

The development of electromyographic prostheses look forward to provide people with the ability to regain the necessary functionality to manipulate objects in their daily environment. For this, it is necessary to process the EMG signals to extract their characteristics and carry out pattern recognition.

The use of machine learning algorithms has been frequently reported for the control of upper limb prostheses [5].

Russo et al. [6] uses a commercial artificial hand (Open Bionics) and a commercial muscle sensor (MyoWare) in conjunction with an Arduino Nano board to classify three different hand movements to control robotic hand movement. They have developed classifiers based on Support Vector Machines (SVM) and neural networks using characteristics in the time domain of EMG signals, obtaining a performance of approximately 90%.

In [7] a pattern recognition system was built through two channels of electromyographic signals to identify four postures of the human hand, using the algorithm of the k nearest neighbors (KNN) to classify signals.

The anthropomorphic robotic hand developed in [8] use a classifier based on Extended Associative Memories (EAM) for the recognition of eight gestures of the human hand.

In this document, different algorithms for pattern classification are proposed and analyzed in order to complement and improve the performance of the prototype made in [8] identifying 8 gestures (different from those recognized by the bracelet through manufacturer's software) and comparing the performance of each algorithm.

II. MATERIAL AND METHODS

A. *Robotic hand.*

The robotic hand is based on a master-slave architecture. The master subsystem is made up of a Raspberry Pi connected to the EMG armband developed by Thalmic Labs Inc., in charge of acquiring data and processing it to classify the user's muscle activity.

The slave device is an ATmega328 microcontroller in charge of defining the movements of the robotic hand based on the information processed by the master device [8].

The electromyographic armband has 8 EMG sensors, and is in charge of obtaining the signals from the muscles of the user's arm, carrying out a small processing on the signals prior to their transmission through Bluetooth 4.0 technology.

The Raspberry Pi 3 has the function of communicating with the bracelet and accessing the transmitted information. In this way, the signals from each of the 8 EMG sensors are obtained and the classification is carried out. The gesture, once identified, is transmitted to the microcontroller through the GPIO ports of the Raspberry Pi 3.

The microcontroller receives the information of the recognized gesture and through a database obtains the position in which each finger of the robotic hand must be located to execute the corresponding grasping strategy. Through an output pin and a PWM signal, the microcontroller tells each of the motor controllers the speed and the direction of rotation with which to move.

978-1-7281-8987-1/20/$31.00 ©2020 IEEE

The motor generates the movement and transmits it to the mechanisms of the hand, producing the movement of the fingers.

The gestures carried out by the prosthesis prototype are those shown in Fig. 1.

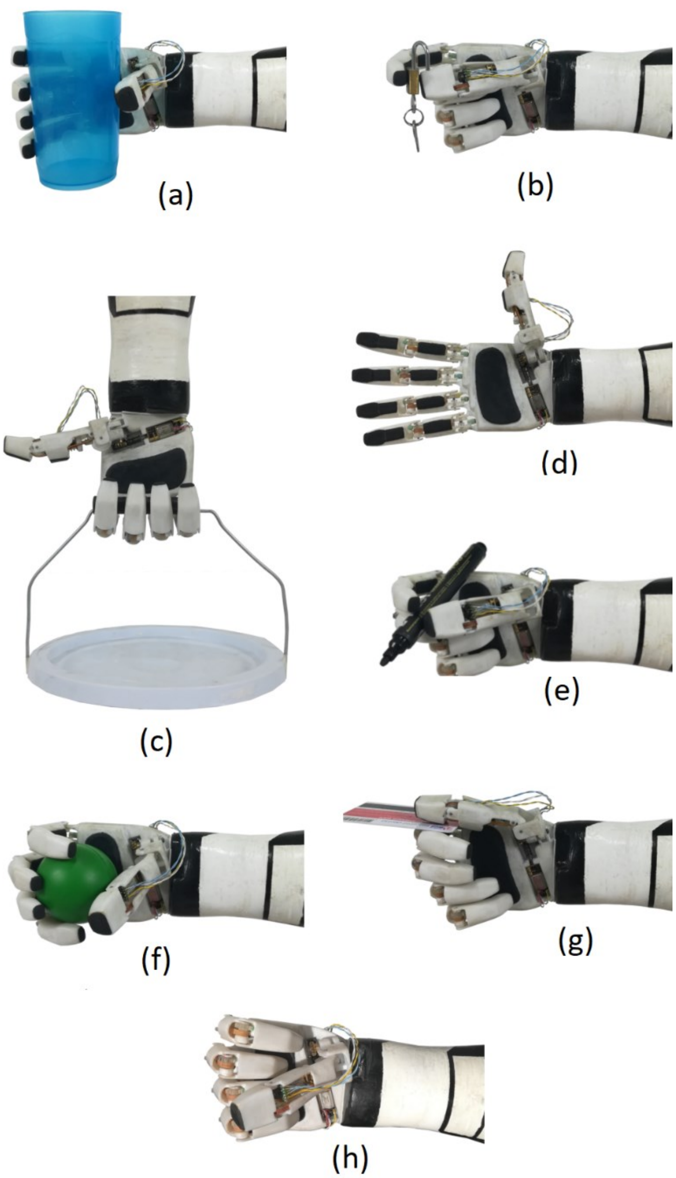

Fig. 1. Types of grasp implemented on the robotic hand: (a) Cylindrical grasp, (b) Tip grasp, (c) Hook grasp, (d) Open hand, (e) Palmar grasp, (f) Spherical grasp, (g) Lateral grasp, (h) Fist.

The classification algorithm used for gesture recognition is based on the Extended Associative Memories (EAM) method, proposed by Sossa et al. in [9].

The purpose of this method is to establish a relationship between an input x with an index μ of the class c_μ and in this way determine to which class a performed gesture belongs.

In [8] it is reported that the performance percentage achieved using this classifier to recognize the 8 defined gestures is 95.83%.

B. *Databases*

The EMG signals with which it works are acquired by the Myo myoelectric bracelet. These signals are sampled with an eight-bit precision at a rate of 200 samples per second.

The bracelet encodes the potential generated by muscle movement to integer values with a range from 0 to 1023, representing the amplified value obtained from each of the EMG sensors [10].

Considering that the cycle period of a gesture is approximately 0.5s, a set of 100 discrete readings obtained from the sensor will represent the gesture performed at any given time [11].

In this way the database available in [12] was generated.

The dataset contains eight files, one for each class for each gesture. Each file contains 50 readings, which in turn consist of 800 samples (100 samples per sensor), forming a base of 40,000 data per gesture.

In Fig. 2 you can see the hand gestures used to generate the classes that produce the different grips on the robotic hand.

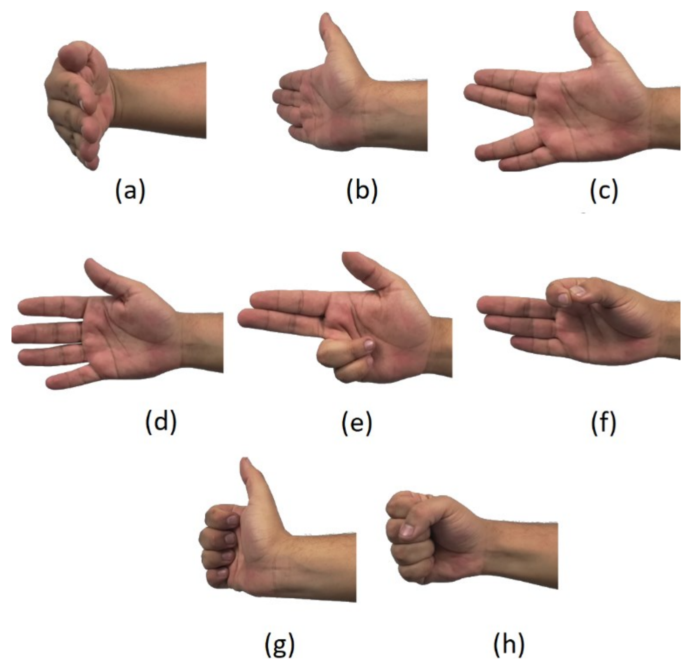

Fig. 2. Hand gesture actions or classes used for generating the grasp types of the prosthesis: (a) Cylindrical grasp, (b) Tip grasp, (c) Hook grasp, (d) Open hand, (e) Palmar grasp, (f) Spherical grasp, (g) Lateral grasp, (h) Fist.

C. *Classification and pattern recognition*

The training and classification were carried out with the database [12] divided into two sets. The first data set consists of 10 random readings of each of the eight gestures (8000 data per gesture) acquired for the training of each classification algorithm. The second data set includes 80 reads (10 different reads per gesture) used to validate and verify the behavior of each classifier.

There are two classification algorithms implemented: a decision tree and Support Vector Machines, each with different variants.

Decision trees are easy to interpret, fast to predict and fit, and require little memory, but they can have low predictive accuracy [13].

Support Vector Machines classify the information by finding the best hyperplane that separates the data between classes, referring to the best hyperplane as the one where the margin between two classes is greater. Support Vector Machines are fast for binary predictions; however, the prediction speed is usually higher for multiclass classifications [13].

To analyze the behavior of the classifiers with the information from the database, the Classification Learner application of the MatLab software was used.

This application allows you to choose between several algorithms to train and validate classification models for binary or multiclass problems [13].

Once trained, validation errors can be compared to choose the best performing model.

The variants applied to the different classifiers are mentioned below.

For the decision tree, three variants were implemented:

Coarse Tree, which uses few leaves to make thick distinctions between classes, using a maximum number of 4 divisions.

Medium Tree, which uses a medium number of leaves to make slightly finer distinctions between classes, using a maximum number of 20 divisions.

Fine Tree, which uses a high number of leaves to make an even finer distinction between classes, using a maximum number of 100 divisions.

For Support Vector Machines, different kernels were used: linear SVM, quadratic SVM and cubic SVM, which perform a linear, quadratic and cubic separation between classes, respectively.

In addition, Gaussian kernel were implemented with different scales: Fine Gaussian SVM, Medium Gaussian SVM, and Coarse Gaussian SVM.

III. RESULTS

The performance of each algorithm for the recognition of patterns in the database can be seen in Table I.

Each percentage represents the quantification of the number of gestures classified correctly by algorithm, corresponding 100% to the 80 data reads used to check performance.

In Fig. 3 the confusion matrix of the Support Vector Machine with cubic kernel is shown, with which the performance of the algorithm can be visualized.

TABLE I
CLASSIFICATION PERFORMANCE PERCENTAGE FOR EACH ALGORITHM.

Algorithm	Performance (%)
Coarse Tree	49.2%
Medium Tree	96.4%
Fine Tree	99.1%
Linear SVM	99.8%
Quadratic SVM	99.8%
Cubic SVM	99.9%
Coarse Gaussian SVM	99.5%
Medium Gaussian SVM	99.7%
Fine Gaussian SVM	99.1%

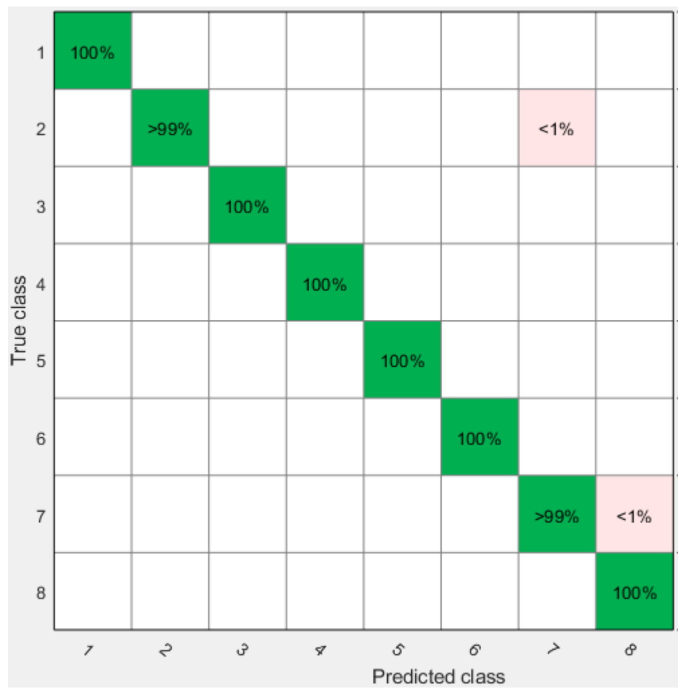

Fig. 3. Confusion matrix for the SVM algorithm classification with cubic kernel

Each column in the matrix represents the number of predictions for each class, while each row represents the instances in the actual class.

IV. DISCUSSION

According to the information presented in Table 1, we can see that the decision tree algorithms have a lower performance than Support Vector Machines, however the fine decision tree can be considered as a viable option as it has a classification percentage 99.1% accurate.

In turn, we can observe that in general Support Vector Machine algorithms respond in a more appropriate way than decision trees, having a higher percentage of performance when linear, quadratic and cubic kernels are used, obtaining the best result with the use of the Support Vector Machine with cubic kernel with 99.9% performance.

From Fig. 3 we can see that the recognition made by the Support Vector Machine with a cubic kernel was not correct in all the tests, showing failures when recognizing gesture 2 when in fact it was gesture 7, and recognizing gesture 7 when the real class was 8. The percentage of each error is shown in Fig. 3 in the red shaded boxes. Despite this, the percentage of incidence of the error in each case is less than 1%.

V. CONCLUSIONS

The use of a single board computer (Raspberry Pi) in the prosthesis prototype developed in [8], gives great versatility to implement various classifiers for pattern recognition and thus improve system performance.

The training and verification of Machine learning algorithms (SVM and decision trees) for classification through MatLab's Classification Learner application is great to test the performance of these algorithms and thus choose the that best results provide the required application.

In this particular project, the decision trees offered lower performance percentages than those obtained with SVM; however, they should be taken into account due to their higher execution speed and ease of interpretation.

The Support Vector Machines algorithms show a very good performance percentage, with the highest percentage being those with the linear (99.8%), quadratic (99.8%) and cubic (99.9%) kernel. One of the advantages that MatLab offers over SVM classifiers is that it is possible to generate the C code of the model trained to perform classifications using the MatLab Coder application, making it possible to export the classifier to run on the robotic hand system.

REFERENCES

[1] J. Vogel, J. Bayer, and P. van der Smagt, “Continuous robot control using surface electromyography of athropic muscles,” in *Proc. 2013 IEEE/RJS International Conference on Intelligent Robots and Systems*, Tokyo, Japan, 2013, pp. 845–850.

[2] Z. O. Khokar, Z. G. Xiao, and C. Menon, “Surface emg pattern recognition for real-time control of a wrist exoskeleton,” *BioMedical Engineering OnLine*, vol. 9, no. 41, pp. 1–17, 2010.

[3] Y. Kim, “The effects of emg-triggered functional electrical stimulation on upper extremity function in stroke patients,” *Physical Therapy Rehabilitation Science*, vol. 2, no. 1, pp. 1–6, 2013.

[4] P. Geethanjali and K. K. Ray, “A low-cost real-time research platform for emg pattern recognition-based prosthetic hand,” *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 4, pp. 1948–1955, 2015.

[5] B. Karlik, “Machine learning algorithms for characterization of emg signals,” *International Journal of Information and Electronics Engineering*, vol. 4, no. 3, pp. 189–194, 2014.

[6] R. Russo, J. Fernandez, R. Rivera, M. Kuzman, J. Lopez, W. Gemin,and R. M., “Algorithm of myoelectric signals processing for the control of prosthetic robotic hands,” *Com Sci Tech*, vol. 18, no. 1, pp. 28–34, 2018.

[7] S. Wan-Ting, L. Zong-Jhe, T. Shih-Tsang, C. Tsorng-Lin, and Y. ChiaYen, “A bionic hand controlled by hand gesture recognition based on surface emg signals: A preliminary study.” *Biocybern Biomed Eng*, vol. 38, no. 1, pp. 126–135, 2018.

[8] L. Sanchez, M. Arias, E. Guzman, and E. Lugo, “A low-cost emg-controlled anthropomorphic robotic hand for power and precision grasp,” *Biocybernetics and Biomedical Engineering*, vol. 40, no. 1, pp. 221–237, 2020.

[9] H. Sossa, R. Barron, and R. Vazquez, “Real-valued patterns classification based on extended associative memories,” in *Proceedings of the Fifth Mexican International Conference in Computer Science*, 2004, pp. 213– 219.

[10] J. Ding, R. Lin, and L. ZY, “Service robot system with integration of wearable myo armband for specialized hand gesture human-computer interfaces for people with disabilities with mobility problems.” *Comput Electr Eng*, vol. 69, no. July, pp. 815–827, 2018.

[11] A. Boyali and N. Hashimoto, “Spectral collaborative representation based classification for handgestures recognition on electromyography signals.” *Biomed Signal Process Control*, vol. 24, pp. 11—8, 2016.

[12] L. Sanchez, M. Arias, and E. Guzman, “Emg data of the myo armband,” Mendeley Data http://dx.doi.org/10.17632/rwbs7645hg.1 (2019).

[13] MathWorks, “Help center: Choose classifier options,” https://la.mathworks.com/help/stats/choose-a-classifier.html (2020).