Classification of Myoelectric Signals Using Multilayer Perceptron Neural Network with Back Propagation Algorithm in a Wireless Surface Myoelectric Prosthesis

Kevin D. Manalo, Noel B. Linsangan, and Jumelyn L. Torres

***Abstract*—The paper focuses on a wireless myoelectric prosthesis of the upper-limb that uses a Multilayer Perceptron (MLP) neural network with back propagation algorithm in classifying electromyography (EMG) signals. MLP Neural network is composed of processing units that have the capability of sending signals to each other and perform a desired function. The algorithm is widely used in pattern recognition. The network is used to train EMG signals and use it in performing the necessary hand positions of the prosthesis. Through programming a Field Programmable Gate Array (FPGA) using Verilog and transmission of data with Zigbee, the EMG signals are acquired, classified, and simulated wirelessly. The signals are classified and trained to produce the necessary hand movements. The corresponding hand movements of Open, Pick, Hold and Grip are simulated through the Zigbee controller. Z-test is used to analyze the data that were produced and acquired from using the neural network.**

***Index Terms*—Field programmable gate array, multilayer perceptron neural network, verilog, ZigBee.**

I. Introduction

Amputees and deformities are some of the medical problems that have been encountered and still being researched on consistently. Prosthetic devices have been developed in order to solve this dilemma. When using a prosthetic arm, different control systems are used.

Surface myoelectric signals give rich information about the neuromuscular activity from which they originate and can achieve a certain movement. Analysis of myoelectric signals has generated useful information used in clinical diagnosis, as well as in control systems for assistive device [1]. The surface electromyography signals are processed and used for the prosthetic devices. Through using the surface electromyography, it is not required to use any physical motion of the body in operating the prosthesis.

In order to gather these signals, a surface myoelectric control is developed. An FPGA-based implementation of the surface myoelectric system is used for the data acquisition system. Studies have stated that FPGAs are reprogrammable and have a large amount of logic blocks that may be used to store data and produce useful logic operations [2]. After gathering the signals, it is needed to be classified to produce the necessary hand movements. Neural Networks may be used to classify and determine the hand positions from the EMG signals. In classifying the EMG signals, it is needed to be extracted first to reduce the length of the input for the Neural Network. It was stated that that the choice of a feature set has a significant influence on the performance of the EMG pattern classifier [3]. Root Mean Square (RMS) method can be applied to extract the features of the raw EMG and be used [4] for the Neural Network. In classifying the signals, a Multi-layer perceptron (MLP) network, a special type of Neural Network (ANN) may be used [5]. It was stated that when it comes to neural network being applied to Myoelectric Signals, most research had been carried out by using a multilayer perceptron. The network contains one hidden layer in conjunction with the back-propagation algorithm. Using MLP network makes it easier for myoelectric signals to be learned by the neurons [6].

In this study, a wireless myoelectric prosthesis is implemented in a field-programmable gate array (FPGA) using a ZigBee controller. Transmitting the EMG signal wirelessly between two points in order to increase the efficiency of controlling the myoelectric prosthesis [7]. The acquired data is classified through the MLP Neural network using back propagating algorithm in Matlab. Matlab provides easy way to design, create and implement neural networks [8]. A Z-test Analysis of the statistical data that were classified using the MLP neural network is produced to verify if the values produced by the neural network and measured EMG data do not have any significant difference.

II. Methodology

This part of the study provides the detailed steps that were performed and the necessary materials and equipment that were used to design the system.

Fig. 1 shows the conceptual framework of the design. The proposed study comprises of three main stages: Amplification and Filtering Stage, Classification Stage, and Prosthesis Control stage that will be connected wirelessly.

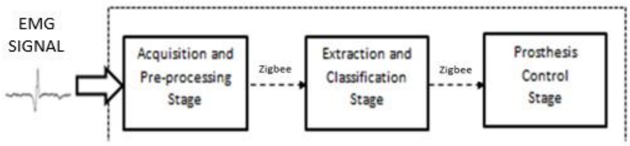

Fig. 1. Conceptual framework of the proposed system.

A. Acquisition and Pre-processing Stage

Manuscript received November 2, 2014; revised March 13, 2015.

K. D. Manalo is with Mapúa Institute of Technology, Manila, Philippines (tel.:+639367473784; e-mail: kevindmanalo@gmail.com).

J. L. Torres and N. B. Linsangan are with Mapúa Institute of Technology, Manila, Philippines (e-mail: jltorres@mymail.mapua.edu.ph, nblinsangan@mymail.mapua.edu.ph).

One of the objectives in the study is to create a Wireless Myoelectric Prosthesis. Fig. 2 shows the block diagram of the Hardware design. The EMG Acquisition will be the first to be performed through using the electrode. The first step to be done is to attach the electrode on the skin surface of the muscles. This signal data will be passed through a signal amplifying and filtering circuit due to a low amplitude and low frequency noise in the raw EMG signal. The resulting analog signal will be converted into a digital signal using the ADC circuit. The signal that is to be classified using Neural Network will be processed into the computer by transmitting it via a ZigBee module which is interfaced in a field programmable gate array (FGPA).

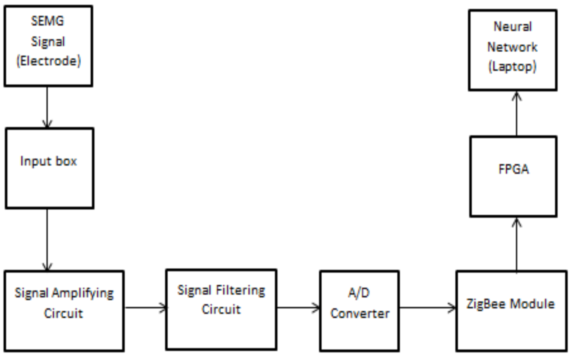

Fig. 2. Acquisition and pre-processing stage.

In developing the software for the design, two programming languages were used: Verilog Programming Language and C# Language. Verilog was used as it is the language compatible in the FPGA. C# is an Object-oriented language that is why it is used to show the data gathered from the EMG sensor. In order to check the values of data before implementing it into the FPGA board, a simulation waveform is created in the Quartus II software. As seen in Fig. 3, different waveforms are produced through the nodes of the EMG main module. After checking if the data is correct and ready for implementation, the Verilog code may be uploaded to the FPGA.

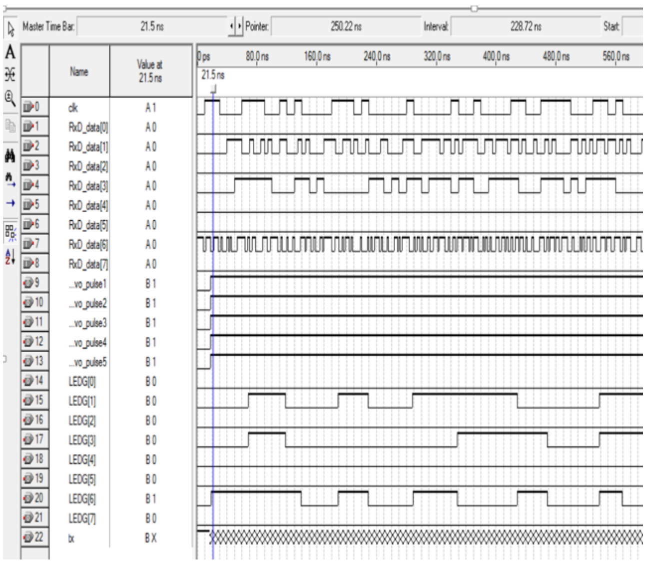

Fig. 3. EMG module vector waveform.

B. Extraction and Classification Stage

In this stage, a process to convert the raw signal into a feature vector is to be implemented. The process will delete the unnecessary noise and highlight important data that is needed for the classification stage. In this study, RMS value feature will be used in reducing the amount of information that will be presented in the Neural Network. RMS has been a commonly used Feature extraction method for its maximum estimation of amplitude in a constant force [4]. The RMS value will be computed and will be used as inputs for classification of the signal. RMS can be stated as follows.

$$RMS = \sqrt{\frac{1}{N}\sum_{t=1}^{N} v_t^2} \tag{1}$$

where v^i is the voltage value at the ith sampling and N is the number of samples in a segment of Equation (1) [4].

For the EMG signals to be classified, it is needed to undergo pattern classification using the Multilayer Perceptron Neural Network which can be performed using the Neural Network toolbox that MATLAB software provides. The resulting class labels and control will be placed in a storage box that will be used for prosthesis testing. As seen in Fig. 4, after the EMG signal has been classified and the feature of the signal has been extracted, the arm movement that will then be controlled by programming the servo motors.

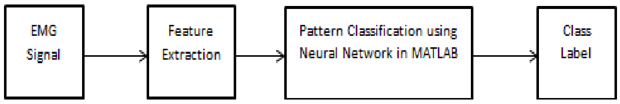

Fig. 4. Block diagram of the classification stage.

In developing the software of the design, two programming languages were used: Verilog Programming Language and C# Language. Verilog was used as it is the language compatible to the FPGA. C# is an object-oriented language that is why it is used to show the data gathered from the EMG sensor. In Fig. 5, the SERVO_UART module is shown. UART stands for universal asynchronous receiver/transmitter. Through this module, a 50MHz clk input, 8-bits binary input, synchronous reset, and transmit input is used to send the data from the FPGA board into the computer. In order for the FPGA and Computer to communicate, signals should pass through the UART.

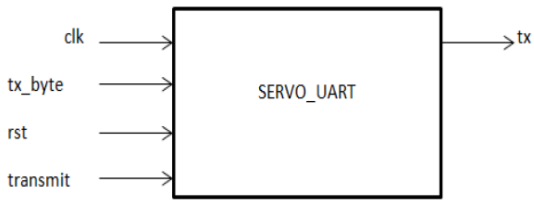

Fig. 5. SERVO_UART sub module.

Designing and Creating a Neural network follows a systematic procedure. Fig. 6 shows the flowchart of the Classification and Testing of the MLP Neural Network using back-propagation Algorithm. The classification will begin by first reading the extracted data and inputting it into the MATLAB software. Parameters and inputs will be specified and used to create the neural network. The next step is to train the neural network. After a specific amount of time, the output of the neural network will be compared to the desired output.

If the experimental result does not comply with the desired output, the training will continue. If the desired output is met, it will be saved and be simulated for the myoelectric prosthesis.

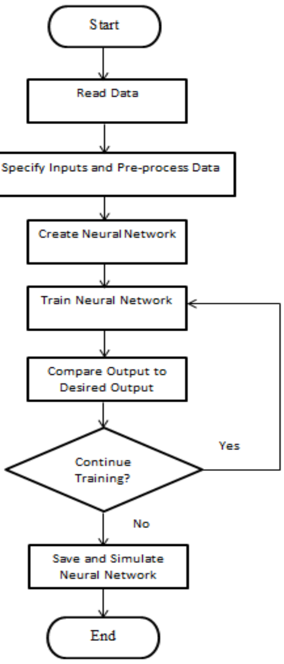

Fig. 6. Flowchart of the MLP neural network implementation.

C. Prosthesis Control Stage

Fig. 7 shows the process on how the prosthesis control stage works. The classified signal from the Multi-layer Perceptron Neural Network with back-propagation algorithm will be transmitted through the ZigBee module. The motor control system will produce the necessary hand position that is checked through the value passed to the system. The result may show the Open, Pick, Hold, or Grip position.

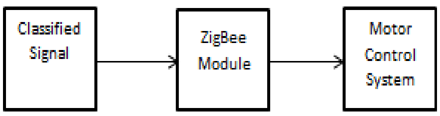

Fig. 7. Block diagram of the prosthesis control stage.

III. Results and Discussion

For this part of the study, it is intended to evaluate the methods that were implemented. The data gathered is evaluated with the usage of Multi-layer Perceptron Neural Network with back-propagation algorithm in classifying the EMG signals.

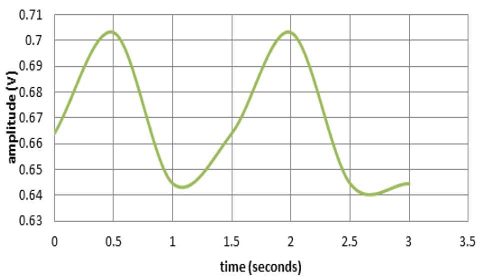

Fig. 8. Measured open signal.

Fig. 8 shows the graph output of the open signal. The value of the signal showed the value is only ranging from almost 0.64V and 0.7V. The value produced has a minimal fluctuation because of the sensitivity in the skin surface.

Fig. 9 shows the line graph of the measured open signal and the signal that is classified through using the MLP Neural Network. The signal has two lowest peaks with 0.665V at 0.5 seconds and 2 seconds. It also has two highest peaks at 0.68V at 1 second and 2.5 seconds. It may be observed that the graph has the same pattern as with the measured open signal.

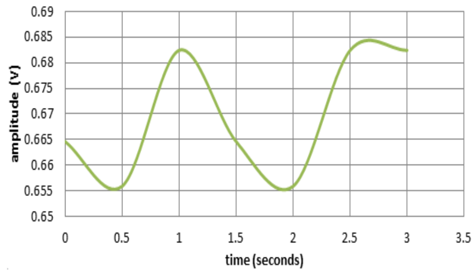

Fig. 9. Classified open signal.

Fig. 10 shows the graph output of the measured pick signal. The value of the signal has a peak of 0.88V at time 3 seconds. Furthermore, it was observed that the lowest point of the signal is 0.76V at time 1.5 seconds.

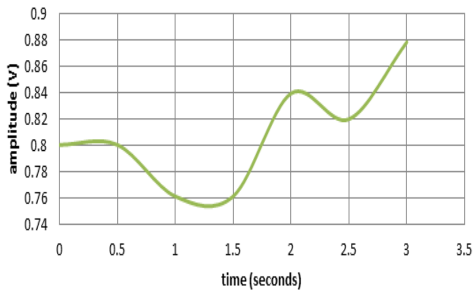

Fig. 10. Measured pick signal.

Fig. 11 shows the line graph of the signal that is classified through using the MLP Neural Network. The graph has an almost constant value with the highest peak of more than 0.9V at 1 second. It may be observed that it the pattern produced is not that fully recognized by the network. It was observed that it was hard to differentiate the signal from the Open signal since the values has increased only a little; therefore, the produced line graph for the pick has also a small difference with the line graph of the open position. Nevertheless, the network was still successful in recognizing the pick position.

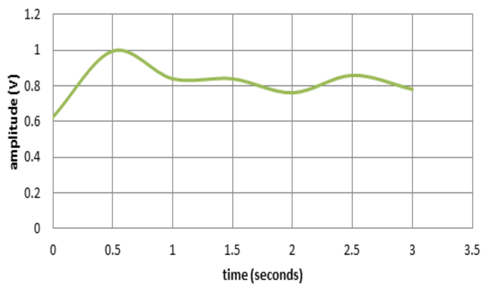

Fig. 11. Classified pick signal.

Fig. 12 shows the graph output of the measured hold signal. The value of the signal showed an almost constant output from 1 second until 3 seconds. It was also observed that at this position, the peak point of the signal is 1V at 0.5 seconds.

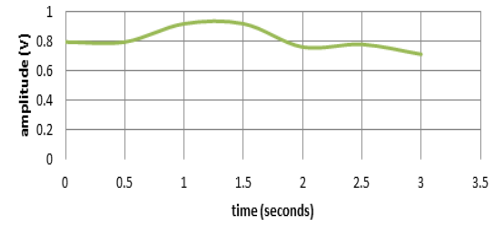

Fig. 12. Measured hold signal.

Fig. 13 shows line graph of the measured hold signal and the signal that is classified through using the MLP Neural Network. The resulting graph shows a constant output of 0.9V from 0 second to 3 seconds. The line graph of both the measured and classified signals have the same pattern based on the values from 0 second to 3 seconds. The pattern was easily recognized by the network since the line graph measured is almost a straight line. The training of the system was fast in the Hold position.

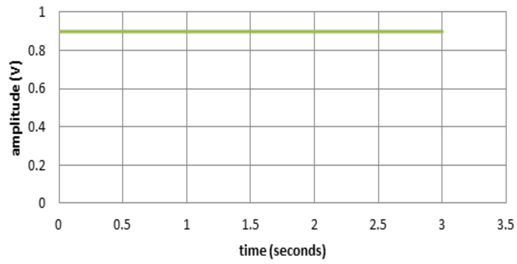

Fig. 13. Classified hold signal.

Fig. 14 shows the graph output of the measured grip signal. The lowest peak of the signal is 0.6V at 2 seconds and has two highest peaks with a value of 1.2V at 1 second and 2.5 seconds, respectively. It may be observed that the average amplitude of the voltage for the grip position has increased due to the force exerted in producing the hand position.

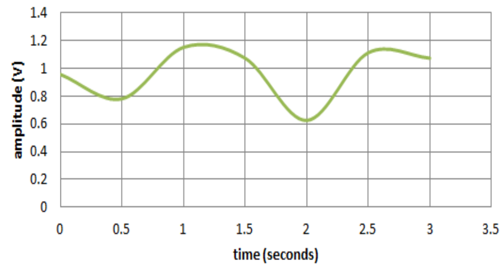

Fig. 14. Measured grip signal.

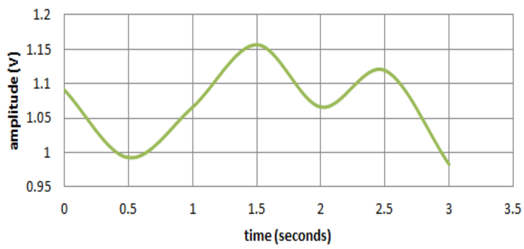

Fig. 15. Classified grip signal.

Fig. 15 shows the line graph of the classified grip signal and the signal that is classified through using the MLP Neural Network. It may be observed that the values from the measured and the classified have produced the same pattern for the grip signal. The amplitude of the voltage started decreasing for a certain period, and then increased again with time. At time, 1.5, the value of the voltage is at its highest peak with 1.15V. It may also be observed that from time 2.5 to 3 seconds, the amplitude has decreased to the lowest peak of the signal which is less than 1V.

Table I shows the values generated by using MLP Neural Network Using 0.05 as the level of significance, Microsoft Excel computes for the z-score and the *P*-value. In these trials, the null hypothesis is that the sample mean is equal to the hypothesized true population mean. As for the signals, the hypothesized true population means are: Open - H_0: μ= 0.438, Pick - H_0: μ= 0. 652, Hold - H_0: μ= 0. 836, and Grip - H_0: μ= 1.060. The alternative hypothesis is that the sample mean has a high significant difference.

TABLE I: DATA FOR CLASSIFIED SIGNAL

Iterations	Classified Signals using MLP Neural Network			
	Open	Pick	Hold	Grip
1	0.449	0.667	0.852	1.144
2	0.442	0.642	0.852	1.265
3	0.442	0.667	0.805	1.182
4	0.449	0.586	0.805	1.140
5	0.449	0.642	0.772	1.200
6	0.442	0.667	0.808	1.218
7	0.442	0.772	0.808	1.144
8	0.449	0.667	0.852	1.136
9	0.430	0.668	0.852	1.053
10	0.442	0.586	0.808	1.100
11	0.442	0.642	0.808	1.039
12	0.442	0.667	0.808	1.011
13	0.442	0.642	0.815	1.040
14	0.442	0.586	0.852	1.035
15	0.430	0.642	0.808	0.985
16	0.442	0.442	0.808	1.013
17	0.442	0.642	0.808	1.040
18	0.430	0.772	0.805	0.999
19	0.442	0.667	0.808	1.037
20	0.430	0.668	0.852	0.968
21	0.439	0.667	0.852	0.969
22	0.435	0.586	0.808	0.955
23	0.430	0.667	0.808	0.958
24	0.436	0.668	0.808	0.984
25	0.435	0.667	0.808	0.966
26	0.430	0.642	0.808	0.967
27	0.430	0.642	0.852	1.015
28	0.430	0.668	0.805	0.965
29	0.430	0.668	0.852	0.952
30	0.430	0.586	0.772	1.014
z-score	0	-0.565	-1.841	-0.616
P-value	1	0.572	0.066	0.538

Through using *z*-test analysis for a single sample mean in

EXCEL, the values of the z-scores and P values for all the positions (Open, Pick, Hold, and Grip) are computed as shown in Table I. With a significance value of 0.05, the resulting z-score would only accept the null hypothesis if P value is greater than it. It was observed that the resulting P values for all the signals were greater than 0.05, thus, the null hypothesis is accepted which means there is no significant difference. The measured values have produced patterns that were classified successfully in all of the positions by using the MLP neural network. Based on the results gathered, the signals were classified successfully. The patterns that were trained for each basic hand position – open, pick, hold, and grip signal, were all successfully classified by the Neural Network.

IV. Conclusion

The study showed that through Multi-layered Perceptron Neural Network with back propagation algorithm using MATLAB, the EMG signals were classified into its corresponding hand position. With 100% for open, 100% for pick, 80% for hold, and 100% success rate for grip, it can be concluded that the EMG signals gathered were classified effectively through the learning algorithm provided by MATLAB. The signals were trained by providing the number of layers, its corresponding nodes, and the classified algorithm which is the back propagation algorithm to produce the network for each signal. Since there is no any significant difference with the measured and the classified signals, it may be concluded that the EMG signals gathered were classified effectively. The signals were tested and verified in the created network by simulating the values in the graphical user interface (GUI) created through C# Language with integrated MATLAB.

Acknowledgment

The authors would like to thank the people who contributed in this study. This research would not be possible if not through the Department of Science and Technology's Engineering Research and Development for Technology Scholarship (DOST-ERDT). The funding and the support were essential in accomplishing all the objectives needed for this study. Also, the authors would like to thank the people who have helped in the testing of the prototype. They have contributed in order to gather the important data needed for this research.

References

[1] S. Herle and S. Man, "Processing surface electromyographical signals for myoelectric control," in *Rehabilitation Engineering*, T. Y. Kheng, Ed., InTech, 2009.

[2] P. Chrapka, "EMG controlled hand prosthesis: EMG classification system," *EE 4BI6 Electrical Engineering Biomedical Capstones*, vol. 59, 2010.

[3] L. Chen, Y. Geng, and G. Li, "Effect of upper-limb positions on motion pattern recognition using electromyography," in *Proc. 4th International Congress on Image and Signal Processing*, 2011, pp. 139-142.

[4] R. Gopura and K. Kiguchi, "An exoskeleton robot for human forearm and wrist motion assist- Hardware design and EMG-Based controller," *International Journal of Advanced Mechanical Design, Systems, and Manufacturing*, vol. 2, no. 6, pp. 1067-1083, 2008.

[5] A. Fougner, "Proportional myoelectric control of a multifunction upper-limb prosthesis," Norwegian University of Science and Technology, Trondheim, Norway, 2006.

[6] P. Parker, P. Englehart, and B. Hudgins, "Myoelectric signal processing for control of powered limb prostheses," *Journal of Electromyography and Kinesiology*, vol. 16, pp. 541-548, 2006.

[7] P. Lichter, E. Lange, T. Riehle, and S. Anderson, "Rechargeable wireless EMG sensor for prosthetic control," in *Proc. International Conference of the IEEE EMBS*, 2010, pp. 5074-5076.

[8] M. Al Shamisi, A. Assi, and H. Hejase, "Using MATLAB to develop artificial neural network models for predicting global solar radiation in Al Ain City – UAE, engineering," in *Education and Research Using MATLAB*, A. Assi, Ed., InTech, 2011.

Kevin D. Manalo is a graduating student of the Computer Engineering (CpE) Department at Mapua Institute of Technology, Manila, Philippines. He is currently pursuing his master's degree in computer engineering (MSCOE) in the same institute. He is expected to finish both degrees in March 2015.

Jumelyn L. Torres is a faculty member of the Computer Engineering (CpE) Department at Mapua Institute of Technology, Manila, Philippines. From this institution, she finished the bachelor of science in electronics and communications engineering (BSECE), graduated as a silver medalist in 2000 and also finished master in science in computer engineering (MSCOE) in 2013. She joined the teaching industry in 2002 right after passing the licensure exam for BSECE.

Noel B. Linsangan is the present program chair of the Computer Engineering (CpE) program at the Mapúa Institute of Technology, Manila, Philippines. Mapúa is the largest engineering school in the country. He obtained his baccalaureate degree in computer engineering from the same school in 1988 and his graduate degree, master in engineering in computer Engineering from the Pamantasan ng Lunsodng Maynila (University of the City of Manila) in 2000. Right after graduation from college he joined the faculty of the computer Engineering program and in 2000 he was appointed as its program chair. He is part of the team that pioneered the implementation of OBE in Mapúa and also one of the point persons of the Institute when its EE, ECE, and CpE programs were preparing for ABET accreditation.