Myo Controlled Robotic Hand

Zackary Minshew
Dept. of Electrical & Computer Engineering
Georgia Southern University
Statesboro, GA, USA
zm00980@georgiasouthern.edu

Jheric Byrd
Dept. of Electrical & Computer Engineering
Georgia Southern University
Statesboro, GA, USA
jb18063@georgiasouthern.edu

***Abstract*—This project explores the use of the surface electromyographic sensors (sEMG) in the Myo® Armband by Thalmic Labs, along with the LattePanda Delta ultimate mini-board by DFRobot and a feed-forward pattern recognition neural network to control a 3D printed robotic hand. To have the previous mentioned items all coordinate together with ease a user-friendly application was created. The application allows for a more welcoming approach for other individuals to use this technology, rather than having to learn the entirety of the complex project architecture. The Myo armband has eight evenly distributed sEMG sensors that read data when the sensor is touching skin. The artificial neural network was initially trained just for five gestures: Rest, Fist, Fingers & Hand Spread, Wave Hand In, & Wave Hand Out. The goal is to expand these five gestures into at least fifteen gestures. These new gestures would be comprised of the previously mentioned five gestures and then some gestures from the American Sign Language Alphabet. The classified output from the artificial neural network is what determines the command for each of the servos in the robotic hand. This document demonstrates how individuals can achieve similar results to that of advanced technology while saving a significant amount of money.**

Keywords—Human Prosthetic Control, LattePanda, Gesture Classification, Myo Armband, Artificial Neural Network

I. Introduction

In today's society it is apparent that the role and use of robotics and process automation is becoming increasingly more prominent in various fields and applications, from self-driving automobiles to various types of manufacturing. The expansion of these fields is driven by the persistent insatiable consumer market, not just in the United States, but in the world. This desire of advancement is typically looking for a means to reduce work or cost. Such as the advancement of robotics in manufacturing that leads to reduced labor and the liability of that labor. A byproduct of reducing the amount of human labor is usually higher profits and lower liability (insurance and safety). Another desire for these advancements is to reduce error, which could be done by a more advanced autonomously driving vehicle; or to be able to provide someone who has lost a limb with a fully functional prosthetic that emulates a human limb very well.

An electromyographic signal is the translation of the electrical signals that are transmitted from the motor neurons (nerve cells). The motor neurons send electrical signals to the respective muscle(s) causing it to contract or relax. The EMG signal is a means for diagnostics of the command signals for and individuals' muscles. This information is instrumental in helping doctors diagnose and help patients. There are two different ways to acquire the data of electromyographic signal. One method is use sensors that are placed on the surface of an individual's skin, hence surface electromyographic sensor. The other, more invasive method is to puncture the skin with needles and to stick the needles into muscle tissue to record muscle activity [1].

For this project the Myo armband was used to acquire the surface electromyographic signals. It is a useful tool because it allowed for advanced data collection with a simple means of obtaining that data. The armband has eight surface electromyographic sensors evenly displaced. The signals from these sensors on the armband can be accessed and obtained from a computer via Bluetooth low-energy communication protocol.

Artificial Neural Networks are computerized systems that model the neural network structure of animalia brains. Artificial neural networks can be used for the recognition of patterns, prediction of future events based off past trends, and for the classification of data and information by learning from historical data [2]. For this project the Deep-Learning Toolbox in MATLAB is used to design, train, and classify the selected hand gestures.

II. Components and Materials

A. Hardware

a) Myo Armband: The Myo armband (Fig. 1) is what allowed for the collection and interfacing of the electromyographic signals. The armband is comprised of eight non-invasive surface EMG sensors (sEMG) that detect electronic signals generated by commands from the motor neurons calling for muscle contraction and an inertial measurement unit (IMU). The inertial measurment unit is a device that reads the orientation of the armband and the angular rate using accelerometers and gyroscopes together.

Fig. 1. Myo Armband by Thalmic Labs.

The data from each sEMG sensor and the inertial measurment unit is sampled 200 times per second, meaning a frequency of 200 hertz. This is the frequency that the armband communicates to the bluetooth low-energy dongle that is plugged into a computer's universal serial bus port.

b) 3D Printed Robotic Hand: The artificial robotic hand and arm (Fig. 2) was designed to resemble a human extremity and reenact its motion capability. Based off a design from InMoov, the hand was 3D printed with acrylonitrile butadiene styrene plastic by previous Georgia Southern University Electrical Engineering students. The dexterous hand has motion based off the joints being tensioned with nylon string that is attached to a servo horn, and when the horn rotates it either tightens the string caused the given finger(s) to contract or it rotates the opposite direction causing the finger to extend. There is a servo motor for each finger and one for the wrist. Unfortunately with this design, the motion of the hand's wrist is limited to one plane dimension [3].

Fig. 2. 3D printed robotic hand built by previous Georgia Southern Electrical Engineering Students.

c) LattePanda Delta Board: The LattePanda Delta device (Fig. 3) is going to be used to control the robotic hand, run the deployed artifical neural network, and run the created MATLAB application. The LattePanda Delta deivce is a "Tiny Ultimate Windows / Linux Device". It has an Intel Celeron N4100 8th generation processor, 4gb of LPDDR4 2400MHz RAM, 32gb eMMC memory, and it supports WIFI 802.11 and Dual Band Bluetooth 5.0. The user has the option of three different operating systems: Windows 10, Linux, or Ubuntu. In addition to the standard Intel Celeron processor, the Delta also has an Arduino Leonardo co-processor. Between the device's size, capability of running Windows for the ANN, and the capability of being a microcontroller makes it the ideal component for this project [4].

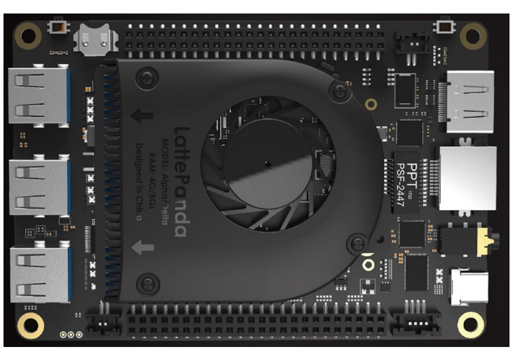

Fig. 3. LattePanda Delta device by DFRobot.

B. Software

a) MATLAB: For this project MATLAB was very instrumental in accessing and analyzing the data, along with creating and training the neural network. The data from the armband sensors was accessed in MATLAB using the Myo-Mex Wrapper [5], meaning that the source code from Myo for the armband was wrote in C++, but to allow consumers to use this device with MATLAB Myo created a wrapper that allows for MATLAB to access, read, and understand the C++ source code of the armband. Using the Deep-Learning Toolbox in MATLAB, an artificial neural network was created that used the collected data from the armband electromyographic sensors as an input to the network. Upon the selection of this input data, the network was trained and configured to be able to assign a classified output based off what input it received.

III. Methodology

Fig. 4. Architecture of the project function and equipment.

A. Experimental Design Process

For the entirety of this semester this experiment was conducted using only two test subjects. One person being a 6'5", 268 pounds, 23-year-old male and the other person being a 6'2", 205 pounds, 23-year-old male. Both subjects executed the data collection process in an identical fashion. Both individuals placed the armband on their right forearm, with the Myo logo positioned on top of the brachioradialis. To record the data from the surface electromyographic sensors that is used to train the neural network, the subjects stood vertically erect with a ninety-degree angle of flexion between their lower arm and upper arm.

B. Signal Conditioning and Data Processing

The artificial neural network required a buffer to be implemented for the data collection process because the startup time was inconsistent which led to variations of signal values in the database. This was implemented by a simple preprocessing task that would run two timed trials of the data collection process upon start up, and then deleting that gathered data because it is not desired.

C. Artificial Neural Network Architecture

An artificial neural network was designed and trained to classify the five original hand gestures. The artificial neural network was a pattern recognition network with eight inputs and five outputs. Once more gestures are permanently added, the amount of outputs will then equal the total number of gestures. For most of the testing, two hidden layers

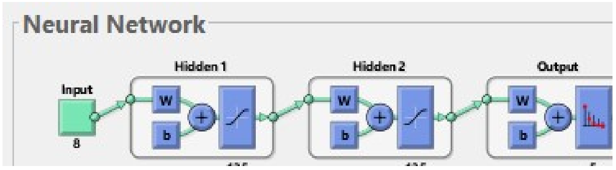

Fig. 5. Artificial Neural Network Architecture

comprised of 125 neurons each were used; along with the epochs being set at 400. There were 120,496 total signals used for the input. For training the artificial neural network, 70% of these signals were used for training while the other 30% was used evenly for validation and training of the neural network. Each input had a target value of one, this is crucial for the gesture classification. The five possible outputs of the artificial neural network were the five different hand gestures.

D. Application Development and Implimentation

An application was created in MATLAB to provide easier interfacing to the armband and artificial neural network. As of now there are three main functions of the app: a data collection process (Fig. 6), building/training the artificial neural network function, and a classified gesture output section that also trends the live data from the surface electromyographic sensors on the Myo armband (Fig. 7).

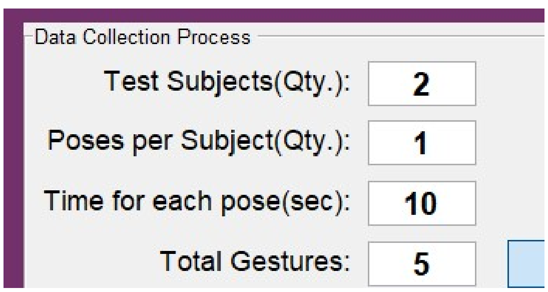

Fig. 6. Data Collection process of the MATLAB application

Once the artificial neural network was built, trained, and tested; a function to stream live data from the sensors of the armband to the artificial neural network was created. The data from the armband was sampled every 350 milliseconds. The classified gesture output (text in red) that is displayed is what will also be the output from the neural network that determines what command to send to the servo motors.

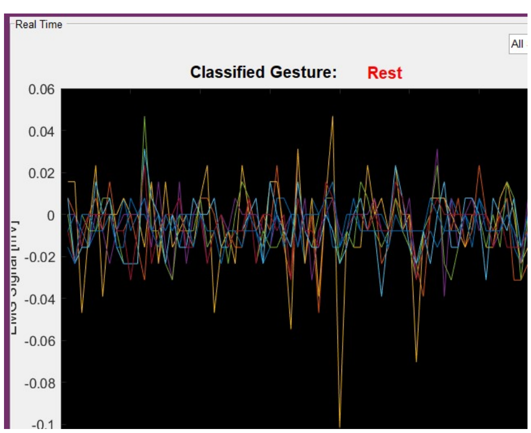

Fig. 7. Live Data & Classification process of the MATLAB application

IV. Results

From the validation check-fail plot of the neural network it can be observed that there were only a handful of validation failures that exceeded two for each individual epoch (Fig. 8). These uncharacteristic errors occurred in the range of 300-350 epochs.

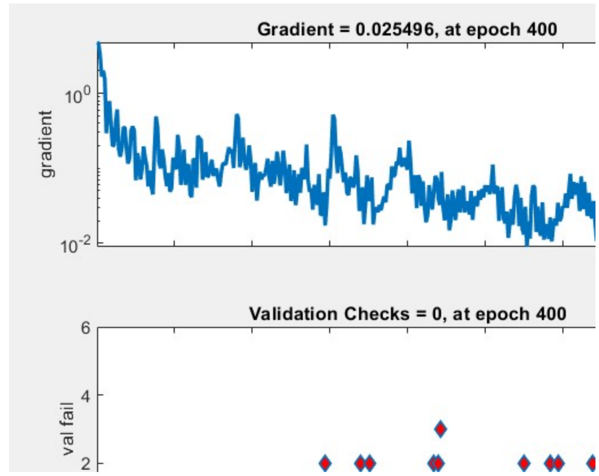

Fig. 8. Validation Fail Check plot of the trained ANN

The performance of the artificial neural network was calculated using the crossentropy function (Fig. 9). This function operates by penalizing errors in classification [6].

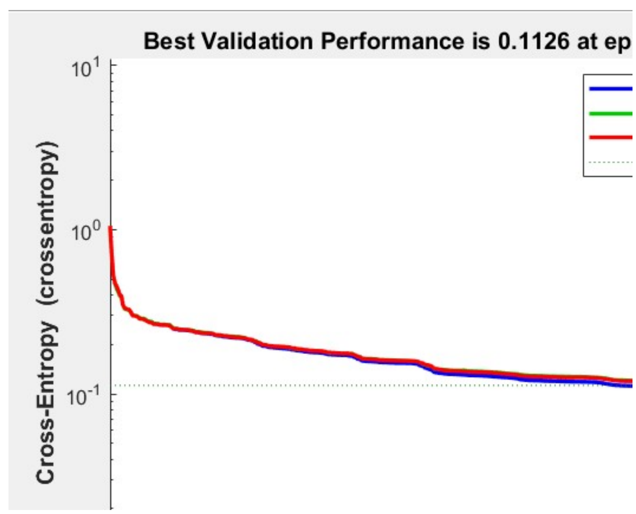

Fig. 9. Performance plot of the trained ANN

The results of the accuracies from the trained neural network are positive results considering two different people were involved and not just one. Adding more people creates more diversity of the data in the database, which leads to greater room for error or confusion by the artificial neural network. The five output classes from class one to class five were the five original gestures: rest position, making a fist, spreading all five fingers out, waving the hand in (flex at the wrist), and waving the hand out.

From the confusion matrix (Fig. 10) it may be observed that the overall accuracy of the trained neural network was 79.5%. Class one (hand in rest position) had a slightly below average accuracy rate at 73.5%. This could be due to the overall magnitude of the signals from the electromyographic sensors is relatively low since there is no major muscle contraction. Class two (making a fist) had a very good accuracy rate of 98.5%. Whereas class three (spreading of the fingers) had a very low accuracy rate of 65.9%. This could be in part of the similarities of the muscle contraction between making a fist and spreading of the fingers. Based off the data in the confusion plot, it can be seen that the output class four was classified as the target of class three and vise-versa. This demonstrates a similarity in the muscle group contraction.

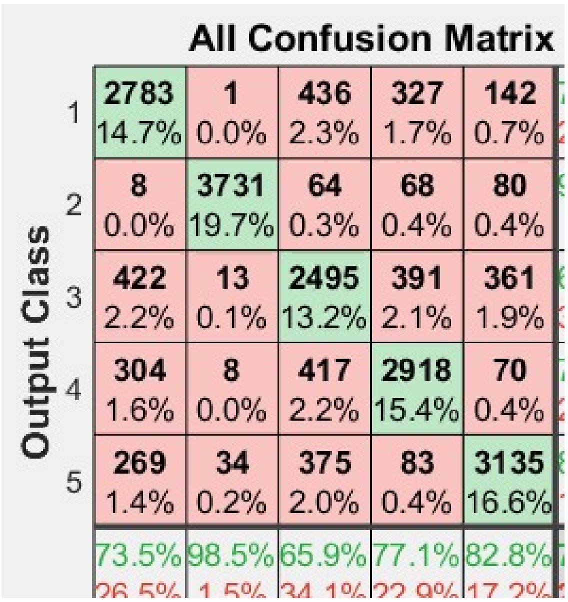

Fig. 10. Confusion matrix result from the trained ANN

V. Conclusions

The hopes of this project are to continue to explore the cost alternative options for human prosthetic control via live human body signals (EMG in this case). Upon the completion of adding more gestures a goal would be to incorporate the use of the inertial measurement unit that has accelerometers and a gyroscope for the sensing components.

References

[1] William Morrison, "Electromyography (EMG)". Internet: https://www.healthline.com/health/electromyography. 2018 [Accessed Dec. 01, 2019].

[2] "What Is a Neural Network?". Internet: https://www.mathworks.com/discovery/neural-network.html, 2019 [Accessed Dec. 01, 2019].

[3] InMoov, Internet: http://inmoov.fr/. [Accessed Nov. 29, 2019].

[4] LattePanda Delta, Internet: https://www.dfrobot.com/product-1910.html [Accessed Nov. 24, 2019].

[5] Mark Tomaszewski, Myo SDK Matlab Mex Wrapper, Matlab File Exchange, https://www.mathworks.com/matlabcentral/fileexchange/55817-myo- sdk-matlab-mex-wrapper, [Accessed Dec. 07, 2019]

[6] " crossentropy" Internet: https://www.mathworks.com/help/deeplearning/ref/crossentropy.html, 2019. [Accessed Dec. 10, 2019].