EMG Based Human Machine Interface
Project Proposal

By Aditya Patel and Jim Ramsay
Advised by Dr. Yufeng Lu and Dr. In Soo Ahn
Published November 29, 2017

Table of Contents

1. Introduction

Electromyography (EMG) is a technique for monitoring electrical signals associated with movement of muscles. EMG signals can be obtained via an intramuscular needle, or by an electrode placed directly on the skin. Intramuscular EMG (iEMG) is more accurate than surface EMG (sEMG) but sEMG allows electrical signals to be measured without the need for intrusive or bulky measurement tools. Acquiring sEMG signals only requires electrodes to be placed on the surface of the skin, directly above the target muscle. When placed on the forearm, sEMG electrodes detect arm muscle activity associated with the movement of a user's hand.

A. EMG Applications

Medical Diagnosis and Rehabilitation

Detection of EMG signals is becoming commonplace in the biomedical field. It is being used in medical research for diagnosis and rehabilitation [1]. In the most common case, an EMG test can be conducted to test for a variety of muscle and nerve related conditions and injuries [2]. Conditions that EMG testing helps diagnose include carpal tunnel syndrome, a pinched nerve, neuropathies, muscle diseases, muscular dystrophy, and Lou Gehrig's disease [3].

Prosthetic Control

In research, EMG signals are used to help recovering amputees control prosthetic limbs. Even if an amputee is missing a limb, their mind can still try to move the limb that is not there. In doing so, electrical impulses are sent to that region of the body as if the limb was still there. For example, an individual missing their forearm can have a prosthetic arm controlled by the EMG signals detected in their shoulder/upper arm [4].

There are great strides being made in EMG based prosthetics. For example, researchers at Japan's Hokkaido University developed an EMG prosthetic hand controller that uses real-time learning to detect up to ten forearm motions with 91.5% accuracy [5]. Additionally, research done at Abu Dhabi University aimed to develop a virtual reality simulation of an arm using EMG signals. They achieved an 84% success rate in simulating the correct movements made by amputees [6].

B. Pattern Recognition Algorithms

Pattern recognition is a subset of machine learning that can be broken into two main categories: supervised and unsupervised. In supervised learning, the algorithm is "trained" by giving the algorithm data that is already classified. This allows the program to have a baseline understanding of the pattern so that it knows what to look for in the future. In unsupervised learning, the algorithm is not given any classification information, and must draw inferences from data on its own [7]. “The most common unsupervised learning method is cluster analysis, which is used for exploratory data analysis to find hidden patterns or grouping in data. The clusters are modeled using a measure of similarity which is defined upon metrics such as Euclidean or probabilistic distance” [8].

A critical part of machine learning is an artificial neural network (ANN). ANN’s are designed to mimic the human brain, where neurons and axons are represented by nodes and wires. Neural networks can be designed in countless different configurations. One form of interest is the Fuzzy Neural Network (FNN) that uses Fuzzy Logic, much like humans. Instead of pure binary decision making, Fuzzy Logic incorporates any value between 0 and 1 to more accurately represent how closely a value matches a set.

2. Problem Statement

The current market for gesture based control of security systems rely solely on the use of cameras to detect user movements. These systems require heavy processing and restrict the user to gesture only in the field of view of the cameras. To address these issues, this project proposes a surface electromyography (sEMG) controlled security system.

There are several practical applications for using an sEMG signal to control security systems. One example is in a small business, such as a convenience store, where an employee would be responsible for monitoring security cameras while working as the cashier. This employee would benefit by being able to use the armband to control the store security camera monitoring system without taking their attention away from the customer. Another example would be if a manager needed to have control of warehouse cameras while working at their desk. The armband would allow the manager to browse through the camera feeds and move the cameras with minimal interruption from their work. One last example is a stay at home mom or dad trying to get work done while a baby sleeps in another room. If this family had an sEMG controlled security system, they would be able to switch between monitoring the baby and checking to see who rang the doorbell without having to touch any buttons or walk to another room. All of these solutions are realizable with the sEMG human machine interface (HMI) security system.

In this project, the user’s gesture is captured by a Myo Gesture Control Armband. It houses eight electrodes for capturing sEMG signals as well as an inertial measurement unit (IMU).

3. Functional Description

A. Functions and Gestures

Function	Gesture	Haptic Feedback
Toggle armband lock/unlock	Fingers spread (hold for 2 seconds)	Vibration (3 seconds)
Calibration Mode	Make fist (hold for 2 seconds)	3 Vibrations (1 second each)
Camera Selection Control Activate	CCW circle with fist	1 Vibration (1 second)
Camera Position Control Activate	CW circle with fist	2 Vibrations (1 second each)
Next Camera	1. Start with palm facing in 2. Move wrist outward	N/A
Previous Camera	1. Start with palm facing in 2. Move wrist inward	N/A
Pan Left	1. Start with palm facing in 2. Move wrist inward	Vibrate low while moving
Pan Right	1. Start with palm facing in 2. Move wrist outward	Vibrate low while moving

B. System Diagram

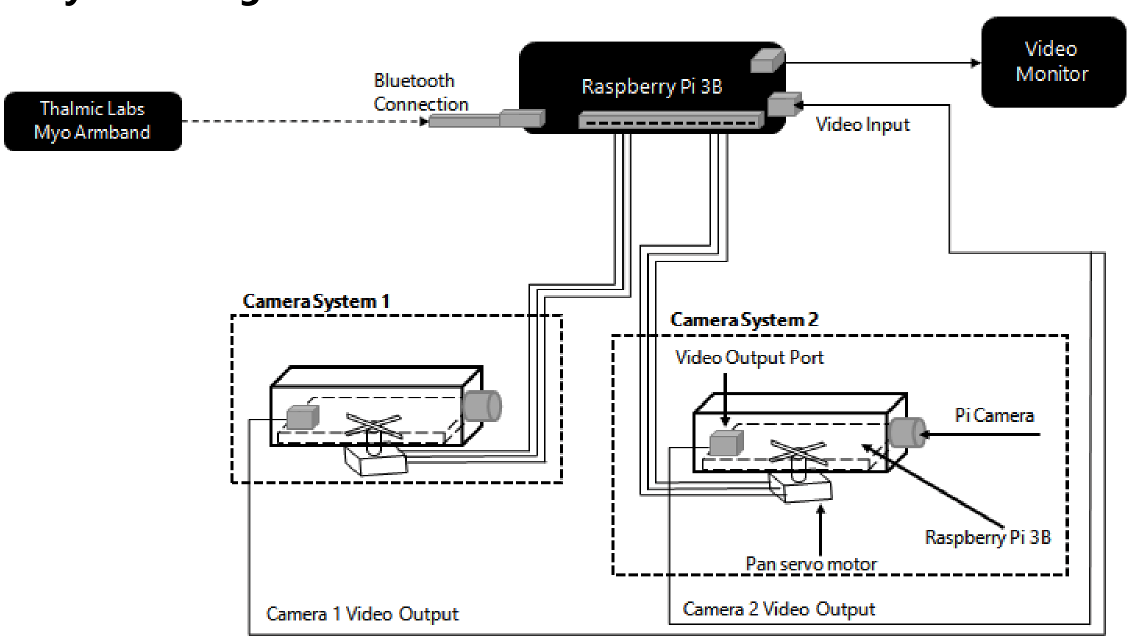

Figure 1: System Diagram

A Myo Armband is worn by a user, giving him/her hands-free control of a video camera system. The user has control of pan, tilt, and camera selection. The system utilizes sEMG and IMU signals from the armband to control the system. The armband wirelessly sends data to an embedded system. The embedded system is responsible for signal processing, control of camera movement and the selection of which video feed to display. Two servo motors are used to rotate each camera. Communications are setup to transmit information between the Raspberry Pi boards, servo motors and the embedded system.

C. Myo Gesture Control Armband

The HMI device used for this project is an sEMG armband, designed by Thalmic Labs. It uses eight sEMG sensors as well as a nine-axis IMU to detect hand and arm movement. Data is sent in real-time via Bluetooth to an embedded system. Instead of reading raw data from the armband, this project uses built-in filters provided by Thalmic Labs. By using the filtered signals, the team can better focus on the gesture recognition algorithms and their accuracy.

D. Embedded System

The embedded system is the heart of the sEMG Security Monitoring System. It receives the armband signal via a Bluetooth dongle. This signal is then processed by algorithms that identify gestures made by the user. The embedded system also generates a PWM (pulse width modulation) signal, which controls the pan/tilt motion of the servo motors. The Raspberry Pi boards transmit the video signals to a communication network where the embedded system will be able to receive the video signals. Based on the user input, the embedded system will transmit the desired video signal to a display for the user to see.

E. Servo Motors

The system includes two pairs of servo motors (per camera). The motors are attached to the case that houses the Raspberry Pi and the camera. The motors are hardwired to the embedded system, which will provide the PWM signals that control their position. By incorporating two motors to the camera mount, the user is able to control both the horizontal and vertical angle of the camera.

F. Raspberry Pi and Camera Assembly

There are two Raspberry Pi 3B computers, each with an attached camera. They process the video and send it to the embedded system across the communication network. The Pi cameras connect directly to the Raspberry Pi 3B and have the ability to stream live video in 1080P, while also recording to an SD card.

G. Monitor

The monitor has three different display modes, one to show all camera feeds at the same time and a full screen mode for each camera. The video feed is sent to the monitor from the embedded system. The selection of display mode is based on the gestures made by the user.

4. Technical Specifications

A. Myo Gesture Control Armband

- ***Physical***
    - Weight: 93g
    - Flexibility: Fits arms ranging between 7.5” and 13”
    - Thickness: 0.45”

- ***Sensors***
    - 9-Axis IMU
        - 3-Axis gyroscope
        - 3-Axis accelerometer
        - 3-Axis magnetometer
    - Made of medical grade stainless steel

- ***Computer / Communication***
    - ARM Cortex M4 processor
    - Wireless Bluetooth 4.0 LE communication
    - Battery
        - Built-in Lithium Ion battery
        - Micro USB charge
        - 1 full day of usage
    - EMG Data
        - Sampling rate: 200 Hz
        - Unitless – muscle activation is represented as an 8-bit signed value
        - Time stamp is in milliseconds since epoch (01/01/1970)
    - Compatible Operating Systems (for the SDK)
        - Windows 7, 8, and 10
        - OSx 10.8 and up
        - Android 4.3 and up
    - Haptic feedback with short, medium and long vibration options

B. Raspberry Pi 3B

- ***Processor***
    - Broadcom BCM2387
    - 1.2 GHz Quad-Core ARM Cortex-A53
    - 802.11 b/g/n Wireless LAN
    - Bluetooth 4.1 (Classic and LE)

- ***GPU***
  - Dual Core VideoCore IV Multimedia Co-Processor
  - OpenVG and 1080p30 H.264 high-profile decode

- ***Memory***
  - 1 GB LPDDR2

- ***Operating System***
  - Boots from Micro SD card
  - Runs Linux OS or Windows 10 IoT

- ***Dimensions***
  - 85 mm x 56 mm x 17 mm

- ***Power***
  - Micro USB socket 5v1, 2.5A

- ***Peripherals***
  - Ethernet
    - 10/100 BaseT socket
  - Video Out
    - HDMI (rev 1.3 & 1.4)
    - Composite RCA (PAL and NTSC)
  - GPIO
    - 40-Pin 2.54 mm expansion header 2x20 strip
    - 27-Pin GPIO
    - +3.3V, +5V and GND supply lines
  - Camera
    - 15-Pin MIPI Camera Serial Interface (CSI-2)
  - Display
    - Display Serial Interface 15-way flat flex cable connector with two data lanes and a clock lane

5. Preliminary Results

A. Raw Data

While collecting preliminary data, our goal was to test the raw armband data to verify that we can see differences in the data when different motions are made. The armband was placed onto the thickest part of the forearm, with sensor-4 on the top of the forearm, and sensors 1 and 8 on the bottom. Two different motions were captured: palm in, wrist action out (wave out) and palm in, wrist action in (wave in).

The first thing we noticed, which can be seen in both Figure 2 and Figure 3, is that there is a distinct difference in the EMG data when the arm muscles are activated. To prove this, we took samples in 10-second intervals and performed the actions in sets of 1, 3 and 5 actions. We can clearly observe the separate actions in each data set.

The second important detail we noticed was that there is a difference between the EMG sensor data when we performed different actions. Figure 2 shows the EMG data when the wrist is moved outward. We can see that the most muscle activation is on sensors 3, 4, and 5. Some action is observed in 2 and 6, while a relatively low amount of action is seen in sensors 1, 7 and 8. Figure 3 shows the EMG data for when the wrist is moved inward. In this case, we see that the most activation occurs on sensors 1, 7, and 8. There is also some activation on sensors 2, 3 and 6, while almost no activation was observed on sensors 4 and 5.

Our goal, moving forward, will be to filter and analyze this data and then implement pattern recognition algorithms. We will be testing more than just the data from one person performing two actions to increase the accuracy of our pattern recognition algorithms.

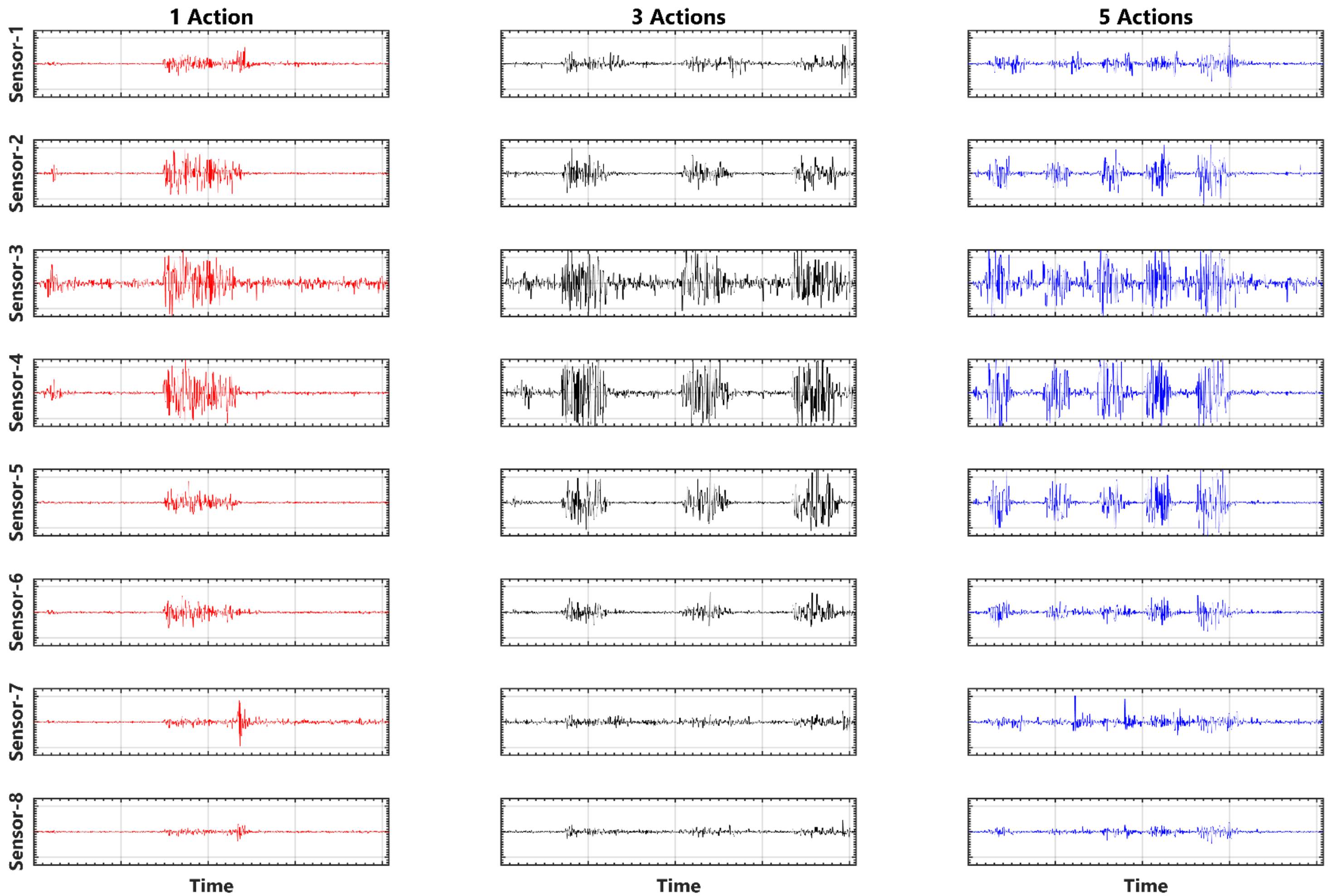

Figure 2: Raw EMG Data with Palm Facing In, Wrist Action Out

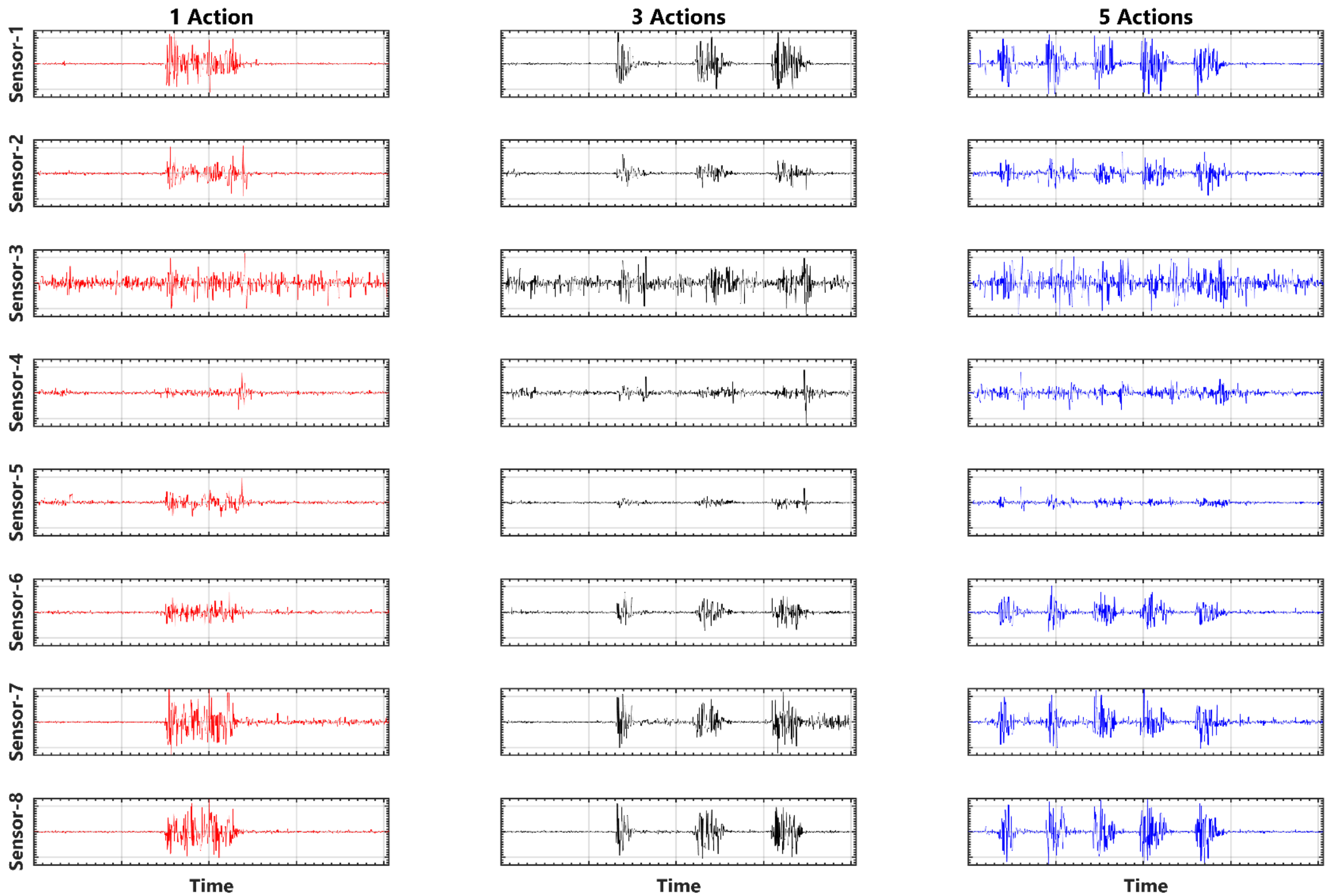

Figure 3: Raw EMG Data with Palm Facing In, Wrist Action In

6. Schedule

A. Schedule of Work

November:

Weeks 1 & 2

1. Write project proposal
2. Develop full parts list and submit order to Chris Mattus
3. Find a way to get the raw data from the armband
4. Develop preliminary filtering methods
5. Revise the website layout. Create a home page.

Weeks 3 & 4

1. Develop project proposal presentation draft
2. Practice presentation
3. Revise project proposal for final submission

December:

Weeks 1 & 2

1. Finalize the website design and post all project deliverables by 12/07/17

Weeks 3 & 4

1. None (Winter Break)

January:

Weeks 1 & 2

1. None (Winter Break)

Weeks 3 & 4

1. Start work on gesture detection
2. Begin development on the Raspberry Pi

February:

Weeks 1 & 2

1. Configure Raspberry Pi and peripherals
2. Set up hardware (monitor, motors, mounts, etc.)

Weeks 3 & 4

1. Finalize and compile code
2. Gather all data needed for final report

March:

Weeks 1 & 2

1. Begin working on final report draft

Weeks 3 & 4

1. Make poster
2. Finish final report draft

April:

Weeks 1 & 2

1. Practice poster presentation
2. Begin drafting project presentation

Weeks 3 & 4

1. Finalize project presentation
2. Practice project presentation
3. Finalize project report

B. Deadlines and Important Dates

November:

11/07 – Project Parts Order

11/09 – Project Proposal Draft

11/28 – Project Proposal Presentation Draft

11/30 – Project Proposal Final Draft

December:

12/07 – Project Website with Deliverables

January – February:

None

March:

03/09 – Student Expo Registration

03/27 – Final Report Draft

03/29 – Student Expo Abstract

April:

04/05 – Poster Printing

04/10 – Student Expo Poster Setup

04/12 – Student Expo Poster Judging

04/13 – Student Expo Award Ceremony

04/17 – Final Presentation Draft

04/19 – Project Demonstration

04/27 – Industrial Advisory Board Poster Session

04/28 – Senior Project Conference

May:

05/01 – All deliverables completed and posted to website

7. Summary

Through the use of surface electromyography, we will develop algorithms that can recognize patterns and differentiate between various hand/wrist motions. With current technology, controlling a system with human gestures is limited. We intend to step up the gesture-based human machine interface industry and develop a security monitoring system that is controlled by a user. The user will wear an armband that will collect and transmit sEMG data via Bluetooth.

Our goal is to make a system where a user will be able to use arm gestures to control which camera feed, in a system of multiple cameras, is displayed on a monitor. The user will also have control of pan and tilt motors to adjust the camera viewing areas.

8. References

[1] M. B. I. Reaz, M. S. Hussain, and F. Mohd-Yasin, “Techniques of EMG signal analysis: detection, processing, classification and applications,” Biological Procedures Online, vol. 8, no. 1, pp. 163–163, Oct. 2006.

[2] “Electromyography,” Medline Plus, 06-Nov-2017. [Online]. Available: https://medlineplus.gov/ency/article/003929.htm. [Accessed: 10-Nov-2017].

[3] J. H. Feinberg, “EMG Testing: A Patients Guide,” Hospital for Special Surgery, 21-Oct-2009. [Online]. Available: https://www.hss.edu/conditions_emg-testing-a-patient-guide.asp. [Accessed: 05-Nov-2017].

[4] S. Sudarsan and E. C. Sekaran, “Design and Development of EMG Controlled Prosthetics Limb,” Procedia Engineering, vol. 38, pp. 3547–3551, Sep. 2012.

[5] D. Nishikawa, Wenwei Yu, H. Yokoi and Y. Kakazu, "EMG prosthetic hand controller using real-time learning method," Systems, Man, and Cybernetics, 1999. IEEE SMC '99 Conference Proceedings. 1999 IEEE International Conference on, Tokyo, 1999, pp. 153-158 vol.1.

[6] L. Fraiwan, M. Awwad, M. Mahdawi, and S. Jamous, “Real time virtual prosthetic hand controlled using EMG signals,” in Biomedical Engineering (MECBME), 2011 1st Middle East Conference on, 2011, pp. 225-227.

[7] C. Donalek, “Supervised and Unsupervised Learning,” Caltech Astronomy, Apr-2011. [Online]. Available: http://www.astro.caltech.edu/~george/aybi199/Donalek_Classif.pdf . [Accessed: 01-Nov-2017].

[8] “Unsupervised Learning,” MATLAB & Simulink. [Online]. Available: https://www.mathworks.com/discovery/unsupervised-learning.html. [Accessed: 14-Nov-2017].