EMG Controlled Bionic Robotic Arm using Artificial Intelligence and Machine Learning

by

Farhan Fuad Rupom
16301122
Gazi Musa Al Johan
16101050
Shafaitul Jannat
16301050
Farjana Ferdousi Tamanna
16301173

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
Brac University
April 2020

Declaration

It is hereby declared that

1) The thesis submitted is our own original work while completing degree at Brac University.
2) The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3) The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4) We have acknowledged all main sources of help.

Student's Full Name & Signature:

Farhan Fuad Rupom
16301122

Gazi musa al johan

Gazi Musa Al Johan
16101050

Shafaitul Jannat
16301050

Tamanna

Farjana Ferdousi Tamanna
16301173

Approval

The thesis titled "EMG Controlled Bionic Robotic Arm using Artificial Intelligence and Machine Learning" by

1) Farhan Fuad Rupom (16301122)
2) Gazi Musa Al Johan (16101050)
3) Shafaitul Jannat (16301050)
4) Farjana Ferdousi Tamanna (16301173)

Of Spring, 2020 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science And Engineering on April 07,2020.

Examining Committee:

Supervisor:

Md. Motaharul Islam

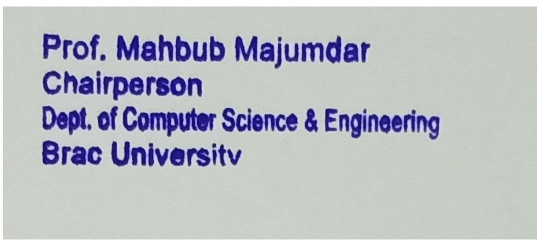
Dr. Md. Motaharul Islam
Professor
Department of Computer Science and Engineering
United International University

Program Coordinator:

Rabiul

Md. Golam Rabiul Alam
Associate Professor
Department of Computer Science and Engineering
Brac University

Head of Department:



Mahbub Alam Majumdar
Chairperson
Department of Computer Science and Engineering
Brac University

Abstract

Electromyography is a unique approach for recording and analyzing the electrical activity generated by muscles, and a Myo-electric controlled prosthetic limb is an outwardly controlled artificial limb which is controlled by the electrical signals instinctively produced by the muscle system itself. Artificial Intelligence and Machine learning is very powerful in every technological field along with biomedical field. The purpose of this work is to utilize the power of Machine learning and Deep learning for predicting and recognizing hand gestures for prosthetic hand from collecting data of muscle activities. Although this technology already exists in the technological world but those are very costly and not available in developing countries. So, designing a cost effective prosthetic hand with the maximize accuracy is the major focus and objective of this work. We have also used a data set recorded by MyoWare Muscle Sensor which represents uninterrupted readings from 8 sensors. We have used Deep learning with the data set for predicting the following gestures which are hand-open, hand-close, spherical-grip, and fine-pinch. Then we used some algorithms of Machine Learning which are K-nearest Neighbor (KNN), Support Vector Machine (SVM), and also the combination of KNN and SVM both for feature classification on data recorded with the 8-electrode surface EMG (sEMG) MyoWare Muscle Sensor. Using the combination of SVM and KNN We have accomplished a real time test accuracy of 96.33 percent at classifying the four gestures of our prosthetic hand. This paper also represents 3D modeling of the robotic hand and its control system using Autodesk 3D's Max software, EMG MyoWare Muscle Sensor, Machine Learning and Deep Learning.

Keywords: Electromyography, Hand gestures, K-nearest Neighbor (KNN), Support Vector Machine (SVM), EMG MyoWare Muscle Sensor, Autodesk 3D Max software, Prosthetic

Dedication

We dedicate our full thesis to our beloved parents.

Acknowledgement

First of all, we want to show our gratitude to Almighty Allah for whom our thesis has been completed without any major interruption.

Secondly, we would like to thank our supervisor Dr. Md. Motaharul Islam for his tremendous support and guidance that helped us a lot to finish our thesis within time and his guidelines always helped us to choose the right way during our thesis. Furthermore, we are grateful for all the knowledge that we have learned during the period of this thesis.

Next to that, we would like to thank our parents for believing in us and without their support it may not be possible. Lastly, we would like to thank all other faculty members, stuffs and few of our friends. They always kept us motivated and helped us by giving mental support and strength for finishing our thesis.

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

$3D$ Three dimensional

ARM Auto Regressive Model

CSV File format

$dist$ Distribution function

EMG Electromyogram

HPF High pass filter

IAV Integrated Average Value

KNN k-Nearest Neighbors

MAV Mean Absolute Value

ms Millisecond

NLP Natural Language Processing

OAA One against all

OAO One against One

RBF Radial Basis Function

RMS Root Mean Square

$SEMG$ Surface Electromyogram

SGN Signum Function

SSC Slope Sign Changes

SVM Support Vector Machine

WL Wave length

ZC Zero Crossings

α	Alpha
ϵ	Epsilon
ω	Omega
Φ	Phi
Σ	Summation

Chapter 1

Introduction

1.1 Thoughts Behind the Thesis

Due to diseases, accidents, and congenital defects people from worldwide are losing their body parts daily. As the road and industrial accidents are increasing in developing countries like Bangladesh, the percentage of disable people is increasing. In a statistics review it has been known that, in the USA, there are already 2.1 million people are existing without their body parts which they have lost in many accidents, unexpected incidents, diseases and the predicted number is going to be doubled by 2050 [1]. Due to limb loss a person's life become more struggling because of facing emotional and financial lifestyle changes. But the good news is being a paralyzed or disable does not mean living without independence in this era when technological world that we are watching today where amputation or any disability can be turned into power for those persons. Therefore, the challenge of this work is to make research and study the different possible ways of using artificial intelligence with prosthetic hands to improve the lives of amputees. But The costs of commercially available myoelectric hands are very high, ranging in price from 2 to 30 Lac [12]. As a result, amputee people from developing countries like Bangladesh are not able to effort this costly bionic robotic prosthesis. So our goal is to build a model in a cost effective way so that poor people can afford our model to make less their struggles in their daily life. Prosthesis types which replaces a lost body parts such as hands, limbs and legs are called an artificial limb. There are mainly four kind of artificial limbs which are Transradial prosthesis, Transhumeral prosthesis,Transtibial prosthesis, Transfemoral prosthesis [14]. We have mainly focused on the transradial prosthesis as it is a prosthetic limb that can take place perfectly of a hand missing under the elbow which is called upper limb prosthesis. While doing this work, the challenge we have taken is to do unlimited research and studies for finding out the different and best possible ways of using Machine Learning and Deep Learning with EMG based prosthetic hands to do make betterment of the amputees. We selected the unique approach electromyogram (EMG) as it is one of the sources that can be used as a control method for a prosthetic hand and simple to use. Surface EMGs passes information about the muscular activity by using those readings as a source of controlling process for prosthetic limbs [24]. As our prosthetic hand is assisted with a sensor system it is 3D printable, affordable, easy and not unpleasant to use. The contributions of this thesis are given below :

- Firstly, we have used a dataset recorded by MyoWare Muscle Sensor which represents uninterrupted readings from 8 sensors.
- Secondly, for feature extraction, Variance, Waveform Length, Integral of EMG, Zero Crossings, Slope Sign Changes, Auto-regressive Model, Integrated Average Value and other techniques are used.
- Thirdly, for feature classification SVM, KNN, and combination of KNN and SVM algorithms are used . The combination of SVM and KNN has the highest accuracy rate among other two algorithms for classifying the four different hand gestures which are hand-open, hand-close, spherical grip, and fine pinch.
- Finally, after identifying different gestures, we have used the classifiers to collect and transfer information and to control our prosthetic arm. We have also used the 3D modeling and rendering software named Autodesk 3D max for implementing our system for the software simulation system.

1.2 Problem Statement

As with the pace of modern science, the biomedical systems and devices have also got advanced in technology and features, but the sector has not gone far enough to provide the instruments which can perform precise data classification detection and act accordingly with low cost production. Even in this era, many people are moving around without hand or legs or other limbs where the problem should have been solved by technology. Many people losing their limbs every day through accident or medical complexity around the globe. Annually approximately 1, 85,000 individuals are decapitated [19]. Insurgents approximately 1,558 individuals lost their body parts in the wars in Afghanistan and Iraq [19]. Approximately 30 percent of those with limb amputation had bipolar disorder [19]. Lifetime cost of medical care for the individuals with hand or leg loss is approximately $509.275 in contrast of $361 million of the individuals without any body part loss [19]. In 2013, clinic expenses totaled $8.7 billion for clients' amputated [19]. Up to 55 percent of people carrying diabetes who already have lower limb amputation would undergo amputation within two or three years of the second leg [19]. Sometime within five years, approximately half of the people with surgical removal induced by cardiovascular problems would die [19]. So as it can be realized the numerous people around the globe are in need of prosthetic limbs with low cost. Again prosthetic limbs with slightly lesser production cost cannot maintain the quality of performing gestures according to users' gestures, as the data classification and acting according to classification is not that much precise. So it's kind of an one way deal, cost can be reduced but preciseness of performance will be affected or the quality preciseness of acting according to users data can be increased but with sacrificing the option of producing it with low cost. As a result, most of the prosthetic limbs are out of the affordable price for most of the general people.

1.3 Aims and Objectives

Our research objective is to create a system where the data and information of user will be acquired and classified precisely as well as to ensure the quality of performing or movement of the prosthetic limb analyzing the data classification as precisely as possible. Most of the cost effective prosthetic limbs can't maintain the synchronization between users' data or signal acquired through EMG sensor and can't act precisely how the user wants the prosthetic hand to be performed. On the other side, which prosthetic arm or limbs do have these kind of quality in synchronization and performance, are not cost efficient or affordable for financially back lagged people. As a result, our purpose is to develop a system where the production cost will be minimized but the data classification and performance with synchronization will be maintained as far as possible. Our data-set for training will be larger and for that, the decision making system of our proposed model will be able to decide suitable action according users' new data values. We aim to implement machine learning and artificial intelligence through using Support Vector Machine algorithm (SVM) and K-nearest neighbor algorithm (KNN) to classify our data and train those data set accordingly. Moreover, we are going to implement the software system through simulation to rectify any kind of bugs and inconvenience. According to that,the user can almost move his prosthetic hand like a real bio hand which can be produced within low cost and greater accuracy.

1.4 Contribution Summary

The summary of the contributions of this thesis are given below :

- Hardware initialization and sensor calibration was done to integrate the EMG sensor.
- For the hardware part the integration was done through an open source software called Arduino.
- The dataset was collected by an online portal.
- EMG signal was preprocessed by using a software called GUMPY which is an open source software.
- Python language was used for applying the algorithms.
- Dataset initialization using Pandas.
- We have trained the dataset using Keras on the top of TensorFlow.
- We have recognized the EMG dataset using KNN-SVM with an accuracy rate of 96.33%.
- Autodesk 3D was used to simulate the gestures.

1.5 Thesis Outline

The rest of the paper has been described in the following manner. Chapter 1 discusses why we need bionic arm. Research objective is presented at Section 1.3 and the Chapter 2 also talks about the literature review. The EMG signal and electrodes placement, characteristics and nature of the signal described in the Chapter 3. Chapter 4 represents the discussion about the data-set acquisition and characteristics. Signal processing with different methods along with feature extraction from signal presented at Chapter 5. Chapter 6 describes about the modeling of data along with the algorithms and kernels which are proposed. The whole system architecture is discussed at the Chapter 7. System implementation with algorithms, feature selections and necessary calculations are discussed in the Chapter 8. User interfacing, visual representation and analysis of system output are described at Section 9.1 and Section 9.2, Section 9.3 and Section 9.4 show the variability of classification rate,error percentage and accuracy rate result. Those features which were not able to meet up by the system for various reasons and modules along with new features that can be added in future is presented in Chapter 10. Ultimately Chapter 11 concludes our paper.

Chapter 2

Related Works

Katsutoshi Kuribayashi et al.[2] proposed using neural networks to monitor the electromyography and form memory material using two methods. Firstly one is EMG signal rectification and integration and the one is the cycle of the neural network. Because of light weight and compact, SMA is used compared to the EMG signal.

Ryait et al. [7] suggested innovations throughout the field of prosthetic limbs.This paper contains three sections, the first one to illustrate EMG signals effects, the next to examine EMG signal on computer simulations, and even the third to describe various virtual hand models using EMG signals.

Netta Gurari et al.[9] recommended the use of positive guidance for foot, arm and fingers. It allows people to connect with the world.Using a manipulation method that simulates an optimal robotic prosthetic, able individuals tapped on surfaces of different hardness by using a gyroscope vibration signal were recorded At the foot of both the tip such vibrating input works better than in the upper limb.

R. Brent Gillespie et al. [12] suggested a process in using a novel form of tactile stimulation including concepts in novel control. Here the electric elbow support is being used in the form with attachment torques across the arm to feedback grasping movements.The EMG stimulation is derived from both the movements of the forearms. This will operate well with and without pressure control in several different situations, employing reticular formation and reticular formation biceps myoelectric indications to represent power, and used to accurately identify artifacts.Via sensory feedback, this significantly increasing. It is applied in 7 individuals who are able-bodied and also the outcome is effective.

Johnny L. G. Nielsen et al.[14] proposed a novel approach done with surface electromyogram to record information from an upper part of limb to generate force using the collateral limb. They measured from the right wrist in multiple degrees of freedom along with different movements. Isometric forces were also recorded by themfrom the right hand in several movements in terms of degree of freedom.

J. Carpaneto et al. [16] proposed using the EMG signal for a physical prosthetic limb. EMG-based stimulation is done by machine learning in the nervous system and the use of implantation of electrodes in the muscles.The vector machine algorithm is being used to predict different types of grip including grasp gestures using both distal and proximal upper limb tissues with EMG activity.

Islam, ArifulAlam, M. et al. [22] recommended the use of support vector machine (SVM) to classify Electromyography (EMG) signals as their designed classifier aims and classify ten individual and combined fingers motion command into one of the predefined set of movements. After extracting features and using those as inputs and a linear SVM for the multiclass classification of EMG signal, recorded reports and validation shows that support vector machine can classify EMG signals accurately with a higher classification rate suitable for designing prosthetic devices [22].

Maamri, Hassen. et al.[24] recommended to classify hand gestures of prosthetic arms by identifying EMG signals with Machine Learning and Deep Learning to predict and recognize hand gestures from muscle activities. It gives an offline test accuracy ranging between 80 % and 90 % by collecting public online Electromyographic (EMG) signals and preprocessing the data, extracting features, and implementing machine learning classifiers and deep learning models. While classic machine learning methods were used with feature extraction on data recorded for the purpose of this project with two amputees accomplishing a real time test accuracy of 75% at classifying individual finger movements [24].

Chapter 3

Data Comprehending

3.1 sEMG Signal Detection

An accurate exposure of individual action in surface Electromyography is an essential topic in the in sector of researches for the hand's motion system, a threshold system has been used to expose muscle's timing on and off, analyzing the Root Mean Square values of reformed signals to that thresholds whose result is based on the framework of noise of each channel's mean power [21]. Furthermore, the voltage reference has been removed after the calculation of the set.

3.2 Electromyography and Signals

A motion starts as a signal from our brain whenever we move one of our skeletal muscles. Then the signal reaches from the spinal cord in the brain to the respective muscle's main motor neuron which is shown in Fig 3.1. It creates an electrical signal down the length of the muscle after activation of the motor neuron by the brain and that signal creates polarization and depolarization which is known as the possibilities for action in the muscle.That action potential expands to the muscle for activating the muscle's motor neuron cells, and the signal activates all the sarcomeres along the muscle's strands and the muscles lease. So the implication of a session on Electromyography is to record that action potential. Electromyography can be used in a various way. The most important use of Electromyography is medical diagnostic. Medical diagnostic let us know about the type of signals the muscles exhibit in case of abnormal muscle activities. Another use of EMG and controlling prosthetic arm is rehabilitation which is the purpose of this paper and also application of this. This is also used in bio-mechanical researches for understanding our brain controls over our muscles and this related studies [20]. Mainly two types of electrodes are usually used for Electromyography, surface and intramuscular. Surface electrodes at Fig 3.2 are very common among other methods such as EKG's and EEG's and also used in this work. As they are used just by sticking to the skin, they are very simple to use. For the readings of signals (SEMG) through skin, they have a metal conductor in

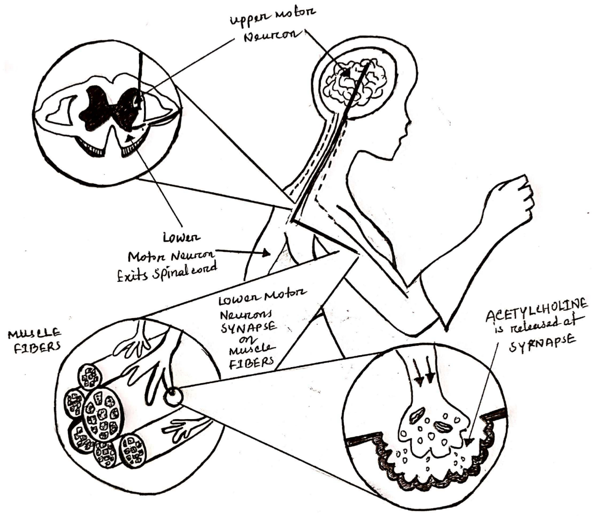


Figure 3.1: Skeleton Muscle Function

the middle with some electric jelly on the other side. They don't cost the patient much discomfort so they are convenient to use. The use of this type of electrodes is as simple as putting them on and taking them off after collecting the readings. They do not need any special training, either. They can't take values from the muscles which are too far below the muscles or too far from the skin because the readings are limited to muscles that are very close to the surface of the skin. They will also take few signals from adjacent muscles. On the other hand, intramuscular electrodes at Fig 3.3 cause discomfort for the patient as they have to be inserted directly into the muscle. This counts as a disadvantage of this type of electrodes. Skin conditions of some people can be swollen by getting a needle stuck in the muscle very significantly. But the advantage of this type of electrodes is they can measure values from any types of muscles as well they also don't get any signals from adjacent muscles.

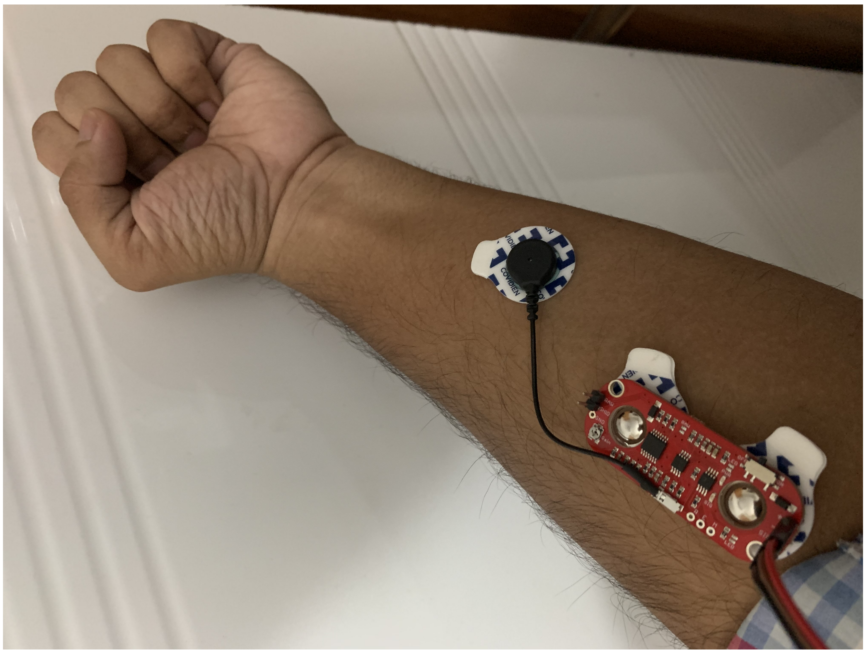


Figure 3.2: Surface electrodes

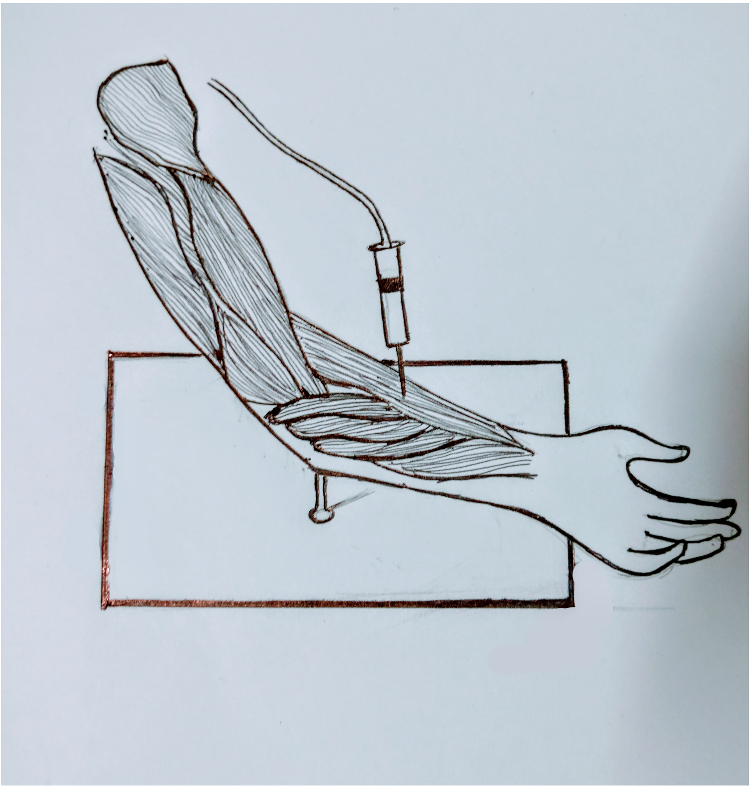


Figure 3.3: Intramuscular electrode

The suggested sEMG signal frequency range of frequency is (5-500)Hz which requires a sampling of frequency more than or equal to 1000Hz [3]. The MyoWare Muscle Sensor is limited to 200Hz. For processing the collected signals accurately robust and adequate techniques are necessary. Prior to amplification, the amplitude range of the EMG signal is 0 mV-10 mV (5 mV) [10]. The raw EMG data contains external, removable noise. The EMG signal is bipolar, which indicates it extends in both positive and negative directions and focuses around zero shown in Fig 3.4.

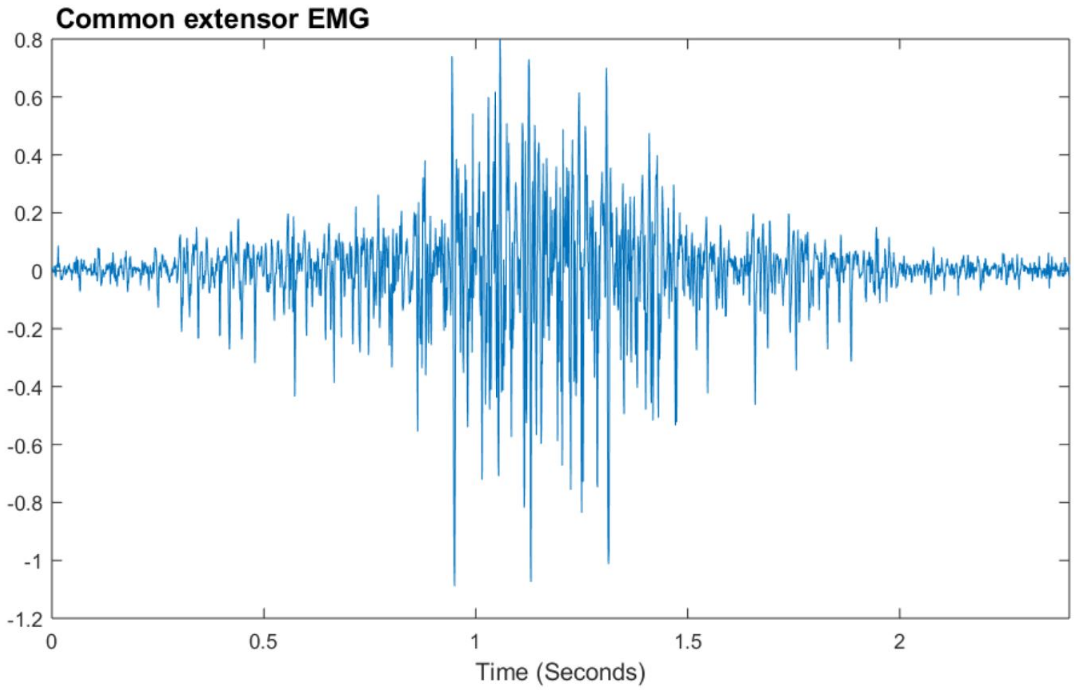


Figure 3.4: Nature of Raw EMG Sign

Chapter 4

Data Acquisition

4.1 Data Collection

Each dataset represents uninterrupted readings from 8 sensors. By this we get EMG data of altogether 64. And the final column stands for the result gesture which is collected by the recording of the given data of class 0 to class 3 [26]. So each line has the following structure of the given below: Data was recorded at 200 Hz,

[8 sensors] [8 sensors] [8 sensors] [8 sensors] [8 sensors] [8 sensors] [8sensors] [Gesture_Classes]

Figure 4.1: Interface of of Data set

meaning that every line is 8*(1/200) seconds = 40ms record time [26]. A classifier can predict a class of gesture (0-3) given 64 numbers. Hand-rest-0, Hand-open-1, Spherical-grip-2, Fine-pinch-3. Fine pinch is the index finger that hits the thumb and extends the rest of the fingers. Spherical grip is like the posture of grabbing a tennis ball. Gestures have been picked very randomly and subjects enlisted were naturally limbed and without physical or neurological problems. They were implanted via the surface electrodes and tested with a sensor EMG (MayoWare). For 20 seconds, each movement was registered six times [26]. Each recording started with the movement already being planned and kept. Recording stopped while still holding the gesture. Through movement is held in a fixed position for a total of 120 seconds [26]. All of them reported in a short time from the same right forearm. That record of a certain type of gesture was concatenated into a .csv file [18] with a name (0-3) at Table 4.1.

Table 4.1: Column Structure of Dataset

Data Source	About This File	Columns
□0.csv 65 *columns*	muscle	$\otimes$+26.0 *muscle reading* 1 *sensor* 1
□1.csv 65 *columns*	activity	$\otimes + 04.0$ *muscle reading* 1 *sensor* 2
□2.csv 65 *columns*	while	$\otimes + 05.0$ *muscle reading* 1 *sensor* 3
□3.csv 65 *columns*	gesture 0	$\otimes + 08.0$ *muscle reading* 1 *sensor* 4
		$\otimes - 01.0$ *muscle reading* 1 *sensor* 5
		$\otimes - 13.0$ *muscle reading* 1 *sensor* 6
		$\otimes - 109.0$ *muscle reading* 1 *sensor* 7
		$\otimes - 66.0$ *muscle reading* 1 *sensor* 8

4.2 Data Properties

These are just a sample of our data set which read muscle activities to classify hand gestures. Here, a human hand muscle activity producing four different hand gestures has been recorded.The sample of our data set shown in Fig 4.2, Fig 4.3, Fig 4.4, Fig 4.5 sequentially :

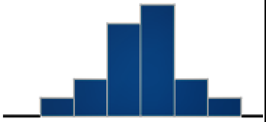
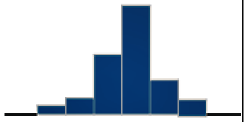
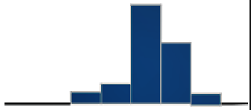
	#26.0 Muscle reading 1 sensor 1	#4.0 Muscle reading 1 sensor 2	#5.0 Muscle reading 1 sensor 3	#8.0 Muscle reading 1 sensor 4
	 -116 111	 -99 90	 -27 34	 -73 55
1	-47.0	-6.0	-5.0	-7.0
2	-19.0	-8.0	-8.0	-8.0
3	2.0	3.0	0.0	2.0
4	6.0	0.0	0.0	-2.0
5	15.0	-5.0	-5.0	-15.0
6	-12.0	-5.0	-1.0	4.0
7	43.0	0.0	-2.0	6.0
8	-26.0	-9.0	-18.0	-60.0
9	-34.0	3.0	9.0	29.0
10	-1.0	0.0	6.0	37.0
11	-10.0	-4.0	-2.0	6.0
12	52.0	1.0	-2.0	-1.0
13	-18.0	-3.0	-4.0	-12.0
14	11.0	0.0	-1.0	8.0
15	35.0	-1.0	-8.0	-12.0

Figure 4.2: Hand-rest(0.CSV)

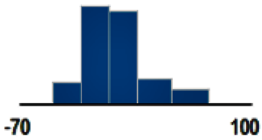
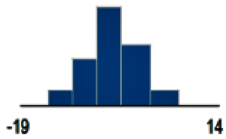
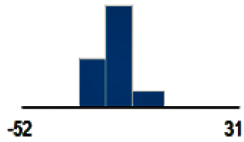
	# -7.0 Muscle reading 1 sensor 1	# -1.0 Muscle reading 1 sensor 2	# -1.0 Muscle reading 1 sensor 3	# 0.0 Muscle reading 1 sensor 4
				
1	-6.0	-2.0	-2.0	-2.0
2	5.0	0.0	0.0	-2.0
3	31.0	4.0	4.0	-2.0
4	-4.0	-4.0	-4.0	3.0
5	-8.0	-3.0	-3.0	0.0
6	-10.0	-7.0	-7.0	-1.0
7	3.0	5.0	5.0	-3.0
8	-1.0	-5.0	-5.0	-4.0
9	-6.0	-2.0	-2.0	1.0
10	-14.0	-4.0	-4.0	-2.0
11	10.0	4.0	4.0	7.0
12	24.0	6.0	6.0	-2.0
13	50.0	3.0	3.0	-3.0
14	-3.0	0.0	0.0	-1.0
15	11.0	2.0	2.0	-2.0

Figure 4.3: Hand-open(1.CSV)

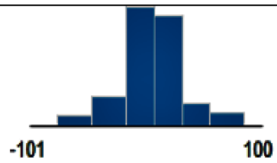
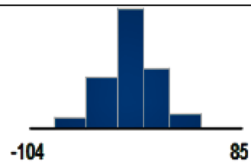
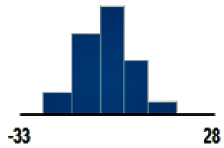
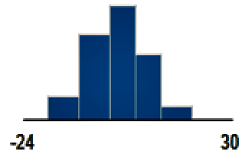
	# 4.0 Muscle reading 1 sensor 1	# 19.0 Muscle reading 1 sensor 2	# -9.0 Muscle reading 1 sensor 3	# -7.0 Muscle reading 1 sensor 4
				
1	-1.0	12.0	20.0	7.0
2	4.0	5.0	-8.0	-2.0
3	-3.0	-3.0	5.0	11.0
4	-5.0	-9.0	-2.0	-5.0
5	3.0	-3.0	-6.0	-5.0
6	5.0	41.0	4.0	0.0
7	-11.0	0.0	-6.0	-9.0
8	24.0	-1.0	19.0	30.0
9	-1.0	7.0	-1.0	0.0
10	1.0	-4.0	-9.0	4.0
11	12.0	-4.0	-2.0	-2.0
12	0.0	-14.0	2.0	-8.0
13	8.0	2.0	6.0	1.0
14	27.0	-2.0	5.0	11.0
15	-21.0	12.0	9.0	1.0

Figure 4.4: Spherical-grip(2.CSV)

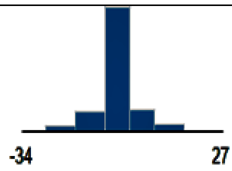
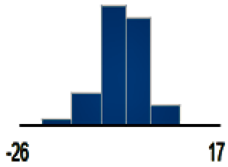
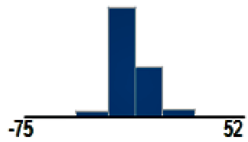
	# -22.0 Muscle reading 1 sensor 1	# -9.0 Muscle reading 1 sensor 2	# -6.0 Muscle reading 1 sensor 3	# -1.0 Muscle reading 1 sensor 4
				
1	-7.0	0.0	1.0	0.0
2	-6.0	-6.0	-6.0	-8.0
3	2.0	1.0	1.0	1.0
4	0.0	3.0	4.0	5.0
5	-11.0	-2.0	-5.0	1.0
6	-20.0	1.0	2.0	-2.0
7	17.0	-2.0	-1.0	-5.0
8	0.0	-2.0	-2.0	1.0
9	-9.0	4.0	6.0	7.0
10	8.0	2.0	0.0	-2.0
11	-15.0	-5.0	0.0	-4.0
12	0.0	0.0	0.0	4.0
13	-5.0	0.0	0.0	-3.0
14	-14.0	-4.0	-3.0	-3.0
15	-12.0	-3.0	-3.0	0.0

Figure 4.5: Fine-pinch(3.CSV)

Chapter 5

Data Devising

5.1 Abstract Signal Processing

In this section, the different commonly used signal processing techniques for EMG signals are explained.

5.1.1 Noise Refining

The purpose is therefore to eliminate any unwanted noise which contaminates the transmission. This seems to be critical because it will effect the data that we evaluate if it has not been eliminated. The majority of EMG transmissions range from 20Hz to 350Hz. [24]. That indicates that a pulse with a wavelength less than 20Hz or greater than 350Hz originates from another point than a muscle [25]. The noise can come from either the electrodes activity, or a particular high frequency transmission from radio frequency propagation and mobile phones [25].

5.1.2 Rectification

Every EMG value would be optimistic after rectification, since all the negative values below the x-axis are folded on the optimistic side after rectified which is defined as full-wave rectification [24]. Another method of rectification is half-wave rectification, whereby the negative values are set to zero rather than being transformed into an absolute value[24].

5.1.3 Normalization

Normalization means transferring a transmission to a level which is related to a certain amount of values. Because each and every electrode plays a part in the output of machine learning, the range of feature values should be standard [24]. Relying on the deep learning technique used, when the importance of any of the variables varies significantly, this feature will overpower through the remainder and adversely affect the output of the program [24]. That is why these functions are standardized such that all the features can make a nearly equal contribution through training [24]. The easiest way to do this is to update the set of functions in the min-max algorithm to new ranges from 0 and 1 or from 1 to 1. The standard distribution is an another option to standardize the data [24].

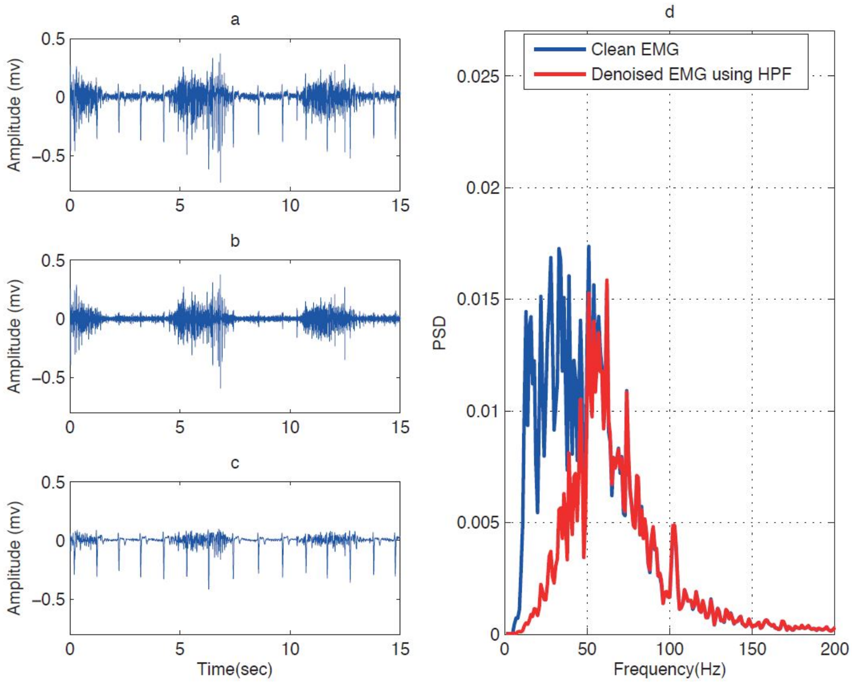


Figure 5.1: (a) Defiled EMG, (b) Deodorized noise EMG, (c) Prediction of noise with filter and (d) Power spectrum density of raw EMG and deodorized noise EMG filter

5.2 Pragmatic Data Processing

It should be recognized that the test wavelength for EMG transmission depends on the number of tests taken in a second, when the testing rate is decreased by analog to digital conversion. The data has used which is tested at 200 Hz and another is at 400 Hz. First of all, data are measured at 200Hz. The EMG transmissions are also filtered as a pre-processing stage by low pass colander along with high pass colander. [24]. Low-pass colanders are those filters, which permit the transmission to transfer in lower frequency data and then amplify the higher frequencies [24]. On the other hand, high-pass filters do the complete reverse; lower wavelengths are eroded [24]. Researchers say in EMG signals that the intention of using a low pass filter is to suit the usual response rate of the muscle. In many other terms, a process delay is generated to imitate the electro-mechanical gap by the low-pass filter [24]. Different Feature Extraction methods have been used before passing the data into another standard machine learning algorithms [24].

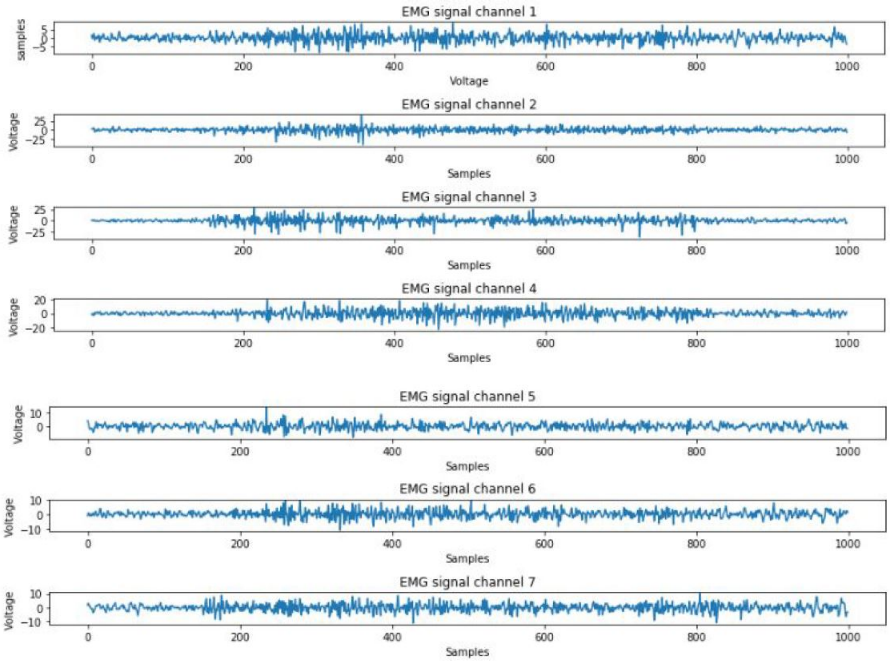


Figure 5.2: Data visualization for an EMG signal on electrodes

5.2.1 Feature Extraction

Variance : It is explained as scope of the sEMG signal's power consistency and is given by [18],

$$\alpha^2 = \frac{1}{N-1} \Sigma_{n=1}^{N} x_n^2 \tag{5.1}$$

Where xn means the data module in the position of nth segment of electromyography pulse including data specimen up to N numbers.

Length of waveform: It is an increasing change in magnitude over the entire time span from sample to sample which implies the caliber of diversity over the sEMG signal. It is done by [18],

$$WL = \Sigma_{n=1}^{N} |x_n - x_n - 1| \tag{5.2}$$

Integral of EMG : The function is an approximation of the addition of sEMG signal infinite values. It is done by [18],

$$I_{EMG} = \Sigma_{n=1}^{N} |x_n| \tag{5.3}$$

Zero Crossings : The function calculates the time the signal passes through zero. This framework is sensitive to sound, so to mitigate noise-induced zero crossing a threshold approach is implemented. It's done by [18],

$$ZC = \Sigma_{n=1}^{N} sgn(-x_n * x_{n+1}) > 0^{|} x_n - x_n + 1| \geq 0.06 \tag{5.4}$$

Slope Sign Changes : The function records the times sign shifts in the slope of the signal. Using this threshold means that the only important variations are calculated to decrease the noise caused by changes in the slope symbol. It's done by [18],

$$SSC = \Sigma_{n=1}^{N} [(x_n - x_{n-1}) * (x_n - x_{n+1})] \geq 0.06 \tag{5.5}$$

Root mean square: It represents mean calculation of signal potency. This is also correlated with the constant intensity and non-tidy constriction. It's done by [24]:

$$RMS \quad = \sqrt{\frac{1}{N}\Sigma_{i=1}^{N} x_i^2} \tag{5.6}$$

Mean absolute value: It provides data about muscle constriction levels [24] . It indicates as:

$$MAV \quad = \frac{1}{N}\Sigma_{i=1}^{N}|x_i| \tag{5.7}$$

Auto-regressive Model: EMG signal has two type of nature which are irregularity as well as unbound characteristic [22]. For these natures, grueling situation arises in signal accurate inspection [22]. Nevertheless for ephemeral gap of time, EMG signal gets entitled to undeviating arbitrary procedure of Gauss. From precursory specimens, time sequence of EMG signal gets exemplified into linear amalgamation for individual specimen that has been represented through [22] ,

$$y_k \quad = -\Sigma_{i=1}^{n} b_i y_{k-i} + g_k \tag{5.8}$$

Here AR is an efficient which is collaborative, the model order along with broadcasting noise is represented through n [22]. Every orders of AR model are employed occasionally but the most used variance is the order of 4th [22]. Catalog of variable framework gets done through the modeling of EMG signal and this is the main benefit of AR model [22]. Classification algorithms can get fed with frameworks in place of utilizing indigenous model of EMG data for the impetus of recalling. For this, computational frailty can get lessened [22].

Integrated Average Value: It is recognized as the aggregated magnitude of non-negative wave signal of a particular wave sign [22]. This term represented as,

$$IAV \quad = \sum_{k=1}^{N} |x_k| \tag{5.9}$$

$$= N.\frac{1}{N}\sum_{k=1}^{N} |x_k| \tag{5.10}$$

$$= \quad N.MAV \tag{5.11}$$

In the above, the signal amplitude is represented by N as well as through a fragment the signal of EMG is indicated with x [22].

The Fig 5.3 is the graphical representation of the distribution of four hand gesture classes after performing feature extraction methods.

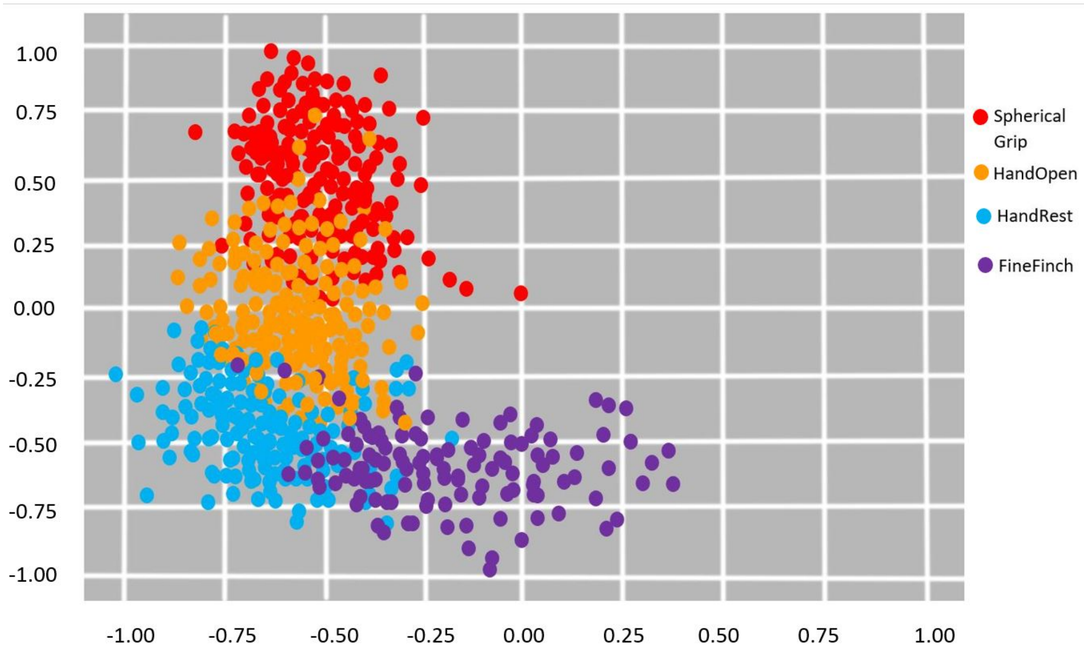


Figure 5.3: Distribution of four hand gesture classes after performing feature extraction methods

Chapter 6

Data Modeling

In our proposed model, we will first collect real data through EMG signal. Then we will do pre-processing and feature extraction of those signals using different methods. After that, we will classify our data set for getting the most accurate value for motion recognizing. Finally, we will feed those values to our simulated model and will test the accuracy of our model.

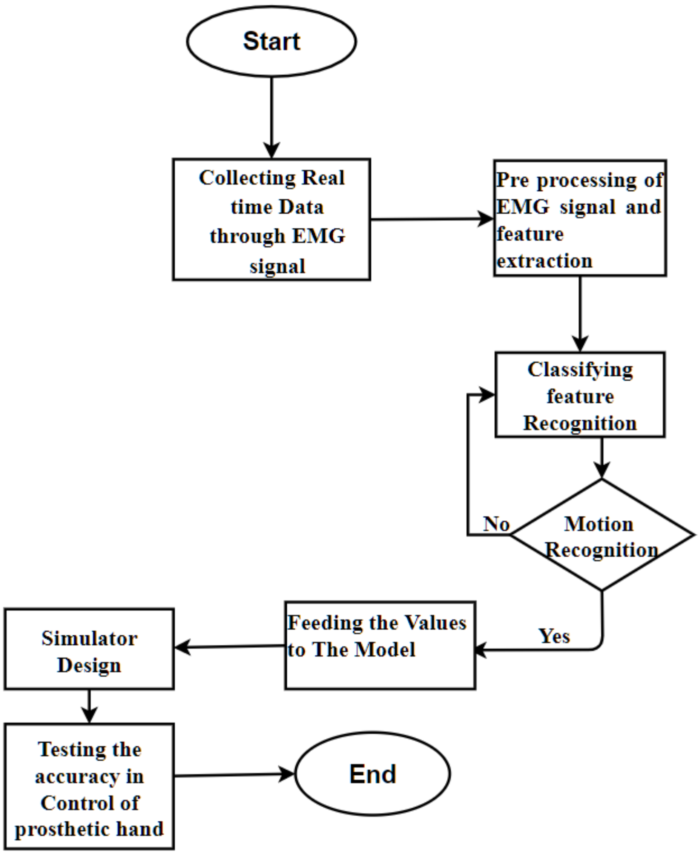


Figure 6.1: Flowchart of Bionic Robotic arm

6.1 SVM

A binary classifier, which determines whether or not a sample is in one class, can be described as SVM. It employs the principle of optimizing structural hazards which compromise the methodological pitfalls and the model's intricacy [15]. The

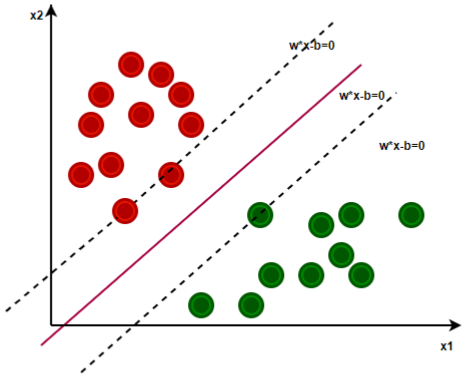


Figure 6.2: Hyper plane with distinguishable data (linear model)

construction of a decision surface is an integral part of an SVM, so as to maximize the distance from positive to negative examples. The judgment surface has a high level, since Rn components are the input vectors. For data which are distinguishable linearly, it can be separated with an optimal separator line [15]. For this the optimal hyper plane for classification can be described by [15],

$$w^T x_i + b = 0 \tag{6.1}$$

where w is a vector of weight and x a vector of input. The ideal hyper plane with two-dimensional linearly separable information shows an example in Fig 6.2. A particular set of input vectors is defined for this hyper plane, known as supporting vectors. This is the ideal hyper plane. Several input vectors are positioned adjacent to the optimum hyper plane. Categorization is then rendered. This is the ideal hyper plane [5]. According to the following terms for any fresh specimens' xi [5],

$$w^T x_i + b \geq 0 \qquad y_i = y_i + 1 \tag{6.2}$$

$$w^T x_i + b < 0 \qquad y_i = y_i - 1 \tag{6.3}$$

Usually Support vector machine (SVM) is a straight forward as well as highly innate concept. In spite of that, when the vector spaces are inseparable, difficulties arise. Problem arises when the data point falls into the space between the ideal hyper planes and in the moment of being the data co-ordinates plotted across hyper planes. A variable i is used to calculate support vector machine calculation that is responsible for changes from the ideal position of the data point. The following optimization problem can therefore summarize the general form of an SVM briefly [8]. Over a range of marked $(x_i, y_i), i = 1....l$ pairs $x_i \epsilon R_n$ and y, SVM the solution is described to [8] :

$$min_{w,b,\epsilon} \quad \frac{1}{2}w^T W + C\Sigma_{i=1}^{l}\epsilon_i \tag{6.4}$$

Variable C presents an inaccuracy attribute that regulates a certain balance among the complexity of an SVM and the number of not separable points. The optimization problem includes the C function, which is a parameter that is user-defined. The user can control the ability of the SVM to generalize using this parameter. This issue is solved by means of quadratic optimization. Since the separation of data is an issue, SVM architecture requires an additional element. SVM's use a nonlinear kernel function which transforms the defined vector into a larger dimension input space of a vector. In this way, the chance of the input vectors being further separated is increased. The operation of a SVM is quite simple after the optimal hyper plane has been built. The vector for input is mapped to the functional area. The SVM then defines the location and judgment of the function in the hyper plane. SVMs' capacity to generalize is their exclusive benefit. With the inclusion of the vector discrepancy in its optimization algorithm as a variable, SVM's can compensate for some input vector heterogeneity which further makes them a good candidate with which to classify EMG.

6.1.1 SVM kernel

SVM incorporates a kernel which organizes input parameters into function compatibility. Three kernels are typically used. There are no strategies to choose a kernel feature to use for the SVM or identify the functionality of the kernel for a specific application. The RBF feature should be first preference. One explanation is that RBF kernel that functions like sigmoid and linear kernels with certain parameters C and Y [8]. A further justification to use the RBF kernel is that numerical computations are less daunting. The RBF kernel throughput fluctuates from 0 to 1, with a polynomial kernel vaguely resembling infinity or 0.

6.2 KNN

K-Nearest Neighbor (KNN) plays a significant role in pattern recognition. This method identifies the homogeneous things that are close to each other. It is mainly used to solve regression and classification problems which need predictions. For ease of interpretation and low calculation time the KNN method gives a very good assumption. KNN classification is used to perform statistical analysis using a discriminant function when stable parametric estimation is quite unknown or determining is tough. This algorithm uses the Euclidean distance between two information (in n-dimensional space two of the given points) is defined by the given formula.

$$dist(x_1, x_2) = \sqrt{(}\frac{n}{i=1}(x_1 - x_2)^2) \tag{6.5}$$

Here, x1 and x2 are two information [11]. It measures the distance between x1 and x2 corresponding to the n attributes [11]. This Euclidean distance is used to get the closeness of the objects in the algorithm. The steps which are used in KNN algorithm are [11]:

1) Determination of the parameters for K.
2) Sorting the distance and finding out the nearest neighbors.
3) Nearest neighbors category Y is gathered.
4) Nearest neighbors simple majority is used.

As our proposed model deals with human motion data we have used this method to get a better result. The electromyography (EMG) signals influences the artificial hand's grabbing and hand grasp opening which is achieved by multiple parameters of different movements of test subjects using KNN rule [11]. Hamming distance or

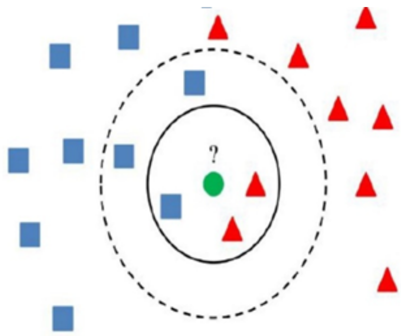


Figure 6.3: K-nearest Neighbor

flap metric can be used here to get better result. In the test sample where the green dot is situated that should be labelled as 1st class and the triangles which are red or the squares which are blue can be labelled as 2nd class. If k=3 then it can be found that two of the red triangles and one of the blue squares is present in the circle which leads to it as second class [23]. And if k=5 that leads to the two of the red triangles and three of the blue squares which declares the first class inside the outer circle [23]. Collection of the trained characteristic vectors along with class tags, this is done by the algorithm's training. Inside the substantial categorization phase the unknown class is regarded as the vector in the marked area. The samples which are close to k are chosen using the respite of different vectors [23]. Detecting the new vector with prediction amongst the K nearest neighbors is one of the most common systems [23]. But this system has a drawback and that is, it depends on the common examples which leads to it's new vector prediction [23]. Hence, this problem can be solved if all the distances of each K nearest neighbors are recognized along with the vectors which are newly categorized and guess of these classes will be based on the distance values [23].

Chapter 7

Architectural Analysis

7.1 System Architecture

Based on the embedded micro-processor structure, the system depends on reasonable and high speed micro-controller is an equating for signal processing and it is convenient for the kind of project needs to inaugurate a higher aptitude along with impressionable and suitable control system. In Fig 7.1 a block diagram demonstrates the versatile control system which is based on the reasonable can be used with the capacity to serve personalized hand postures which suits the lifestyle of the users.

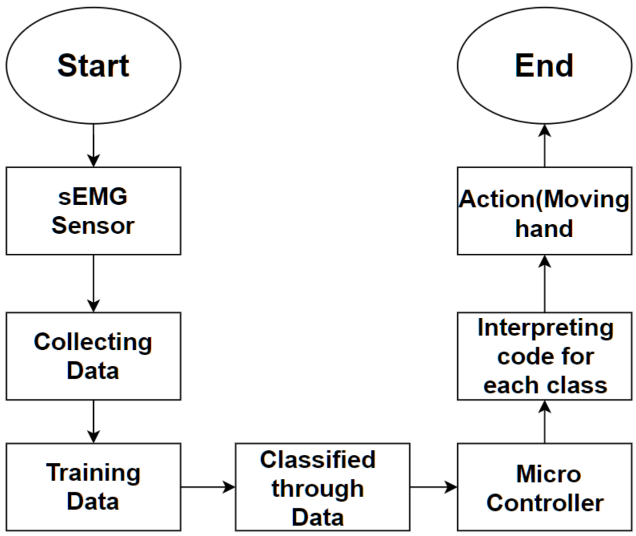


Figure 7.1: The Architecture of System Block Diagram

7.1.1 Physical Architecture

The physical architecture is the representation and layout of a project and its modules in a schema. This architecture indicates the layout of the structure or arrangement of the physical modules of a system. One of its aims to compliment the logical architecture part and project requirements. In this system the module that represents the physical architectures are EMG sensor for taking input; microprocessor, filtering and simulation processor for processing and simulation preview for increasing the accuracy.

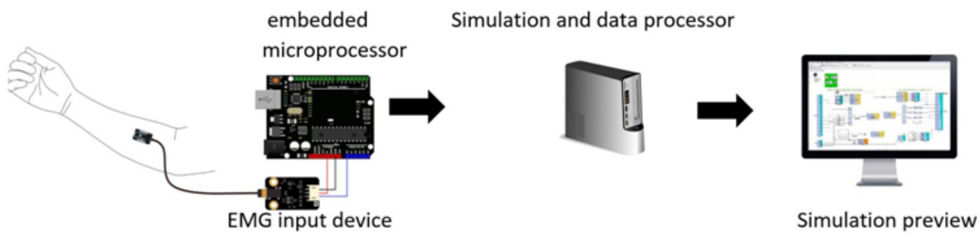


Figure 7.2: Physical Architecture

7.1.2 Logical Architecture

In order to represent the logical machine view, the MVC architecture is used all over [24]. By using this architecture, one can represent distinction among the presentation, the data and the processing. MVC gives three rudimentary fragments whose parts are illustrated :

- Model: Includes all the data for displaying [24].
- View: Encompasses presentation of data to user and interface [24].
- Controller: Controls user inputs and encompasses them to process and sends command for updating model[24].

For interactive application, this kind of architecture is perfect in order to fulfill the requirements [24].

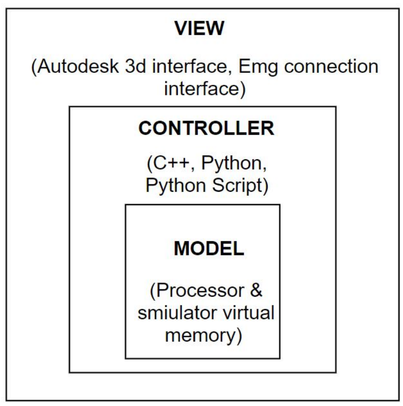


Figure 7.3: Logical Architecture

Chapter 8

System Implementation

8.1 Aperture Size Selection

The data gathered from the EMG is aperture sized, and each window extracts features. The functions derived are often connected to build a broader collection of features. Choosing the best aperture size multiple sliding apertures are selected as 128ms to 1024ms with an interval of 128 aperture and their execution was tested precisely for exactness of EMG signal ramification [22]. The enhancement of the aperture was declared to 128 ms . Ramification of hand activity with each of the aperture sizes has been measured 5 times for each subject. It has then determined the overall ramification rate [22]. Then, it determined the mean of all subject details for every aperture [22].

8.2 Feature Parameters Selection

Classification accuracy is agitated by the possibilities of feature parameters. Table 8.1 determines the time domain traits which are molded into eight feature position for the representation of evaluation. To determine each feature set combined finger movements of individual and the mean sorting of distinct is calculated. The process is repeated for each subject. After that for the certain feature set over all an aggregate classification average rate is found. For the aspect of eight sets the process is iterated with all of the sets inaccuracy or the aggregated classification average rate.

8.3 Feature Vector Calculation

EMG signals sorting mainly lean on the detection of right attribute of vectors which are exclusively for them. Incompatible attribute parameters like ZC, MAV, RMS, WL and etc model are assessed, within each aperture attributes are extracted from EMG data. By using these feature parameters from each EMG data channel features are being calculated. First of all, by choosing the apertures of data then a feature parameter assessed are estimated for that particular aperture. Afterwards sliding the aperture away to review a new aperture of data and then the computation of feature values for each factor parameters is replicated. The computation features from one class of EMG data for example Hand Close of 1st aperture, 50th aperture, 100th aperture and 150th aperture as an example is shown on Table 8.2 and Table 8.3. Concatenation of the calculated feature values are then used to form a substantial attribute vector [22].

Table 8.1: Attribute Sets

Attribute Set	**Attribute Parameters**
First	ARM(order 4), WL
Second	WL, ARM, MAV
Third	MAV, ARM(order 4), RMS,
Fourth	RMS, ARM (order 4), WL
Fifth	RMS, WL, IAV, MAV
Sixth	WL, RMS, ARM (order 4), IAV
Seventh	MAV, ARM ,WL, RMS,
Eighth	ARM, WL, MAV, IAV, RMS

Table 8.2: 1st and 50th aperture attribute parameter values calculation

Attribute Parameter	**Aperture 1**		**Aperture 50**	
	Chn no. 1	**Chn no. 2**	**Chn no. 1**	**Chn no. 2**
MAV	0.000258	0.000451	0.000303	0.000497
RMS	0.000373	0.000562	0.000296	0.000501
WL	0.012994	0.017981	0.014003	0.013027
IAV	0.139863	0.219958	0.141967	0.201129
ARM(order 4)	-1.39889	-1.9117	-2.02136	-3.10024
	0.299758	0.714381	0.809527	3.001324
	0.286628	0.399899	0.340131	-1.20913
	-0.03101	-0.24217	-0.20142	0.121514

Table 8.3: 100th and 150th aperture attribute parameter values calculation

Attribute Parameter	**Aperture 100**		**Aperture 150**	
	Chn no. 1	**Chn no. 2**	**Chn no. 1**	**Chn no. 2**
MAV	0.000312	0.000231	0.000251	0.000221
RMS	0.000398	0.000292	0.000301	0.000247
WL	0.009352	0.006997	0.008125	0.006507
IAV	0.156218	0.12597	0.114012	0.110201
ARM(order 4)	-2.52587	-1.9543	1.80252	-2.924
	2.49896	0.918212	0.656075	3.012529
	-0.91074	0.321091	0.298457	-1.58325
	0.059786	-0.2318	-0.16286	0.271091

Table 8.4: Interconnected Attribute Values

Attribute Parameters	**1st Aperture**		
MAV	0.000258		-1.39889
	0.000451		0.299758
RMS	0.000373		0.286628
	0.000562	ARM (order 4)	-0.03101
WL	0.012994		-1.9117
	0.017981		0.714381
IAV	0.139863		0.399899
	0.219958		-0.24217

Channels computed feature values are interconnected solely underneath one another. Interconnection of feature values of channels for 1st aperture is shown on Table 8.4. Then the feature vector is formed by all the integrated feature values of all aspect frameworks which are solely underneath one another. All 4 aperture feature vectors are shown on Table 8.5 for example:

Table 8.5: 1st-150th, 4 aperture aspect vectors

Aperture 1	**Aperture 50**	**Aperture 100**	**Aperture 150**
0.000258	0.000303	0.000312	0.000251
0.000451	0.000497	0.000231	0.000221
0.000373	0.000296	0.000398	0.000301
0.000562	0.000501	0.000292	0.000247
0.012994	0.014003	0.009352	0.008125
0.017981	0.013027	0.006997	0.006507
0.139863	0.141967	0.156218.	0.114012
0.219958	0.201129	0.12597	0.110201
-1.39889	-2.02136	-2.52587	1.80252
0.299758	0.809527	2.49896	0.656075
0.286628	0.340131	-0.91074	0.298457
-0.03101	-0.20142	0.059786	0.16286
-1.9117	-3.10024	-1.9543	-2.924
0.714381	3.001324	0.918212	3.012529
0.399899	-1.20913	0.321091	-1.58325
-0.24217	0.121514	-0.2318	0.271091

8.4 KNN-SVM

A new approach is introduced the new category that blends algorithms such as support vector machine along with the k-nearest neighbors. The method is based on the classification of SVM along with KNN where only one specific point for each class is picked. The method counts radial length among the feature group. Appropriate super-plane of support vector machine in feature space throughout the class step. The test procedure will be listed on support vector machine when the length is bigger comparatively from threshold calculated previously [4]; k-nearest neighbor algorithm would instead be utilized. In the KNN algorithm, each support vector is chosen for a specific region and length among the experimental samples together with all support vectors is being collated [4]. The experimental modules get distinguished through identifying k-nearest reference neighbor. Computational analysis shows that the combined algorithm not only increases precision in comparison with SVM alone, it can also resolve the issue of choosing the kernel parameter for SVM [4].

8.5 KNN

KNN is admired nowadays because of its processing speed. Not only this simplicity in the process of recognition is really amazing but it increases the classification accuracy [13]. If the test specimen's distance to the super plane which is optimal in support vector machine is not greater than the threshold value computed by SVM, the responsibility falls on KNN to do the classification. In terms of time it can perform really well and has better accuracy in classification. EMG signals recorded from muscles of the user's and activation from these muscles were used in order to control a prosthetic arm. From the point of accuracy, simplicity KNN is very good to be implemented on many types of medical related appliances such as artificial limbs and high accurate robotic appliances where human interacts with robot. But KNN has to maintain a simple rule and that is, its algorithm is fully based on the k data point in the data which it will be trained. There it can predict because of its nearest neighbor. For this reason, the selection of k needs to be handled very cautiously as it has supremacy towards the interpretation of classification [17]. K mainly relies on the set of data and the model it has been exposed to. Province of k is from one to ten [17].

Table 8.6: KNN Standard classification accuracy rate

Data Sub	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Avg. in %**
1st Sub	79	90	87	79	82	93	74	80	92	90	84.60
2nd Sub	93	87	87	87	87	77	73	73	84	84	83.20
3rd Sub	91	95	90	82	78	72	72	72	82	92	82.60
4th Sub	95	75	90	100	90	90	90	90	90	95	90.50
5th Sub	88	93	87	86	85	82	92	90	90	90	88.30
										Avg.	**85.84**

Though the accuracy rate is lesser than SVM, it efficient in certain field where the classification rate of SVM is lower and overall it increases the accuracy of classification by supporting SVM. When the efficiency or feature distance in the plane gets lower for SVM, The KNN code gets the opportunity to run. Here is a glimpse of KNN classification code of our system at Fig 8.1.
Without any due it can be said that KNN algorithm is efficient and fast regarding of its simplicity. Although K-value is the main key point of KNN's performance

```python
#KNN

class KNearestNeighbors:

    def __init__(self, k=3, distance_metric=euclidean):
        """Initialize k value and distance metric used for model."""
        self.k = k
        self.distance = distance_metric
        self.data = None

    def train(self, X, y):
        """Zip labels and input data together for classification."""
        # raise value error if inputs are wrong length or different types
        if len(X) != len(y) or type(X) != type(y):
            raise ValueError("X and y are incompatible.")
        # convert ndarrays to lists
        if type(X) == np.ndarray:
            X, y = X.tolist(), y.tolist()
        # set data attribute containing instances and labels
        self.data = [X[i]+[y[i]] for i in range(len(X))]

    def predict(self, a):
        """Predict class based on k-nearest neighbors."""
        neighbors = []
        # create mapping from distance to instance
        distances = {self.distance(x[:-1], a): x for x in self.data}
        # collect classes of k instances with shortest distance
        for key in sorted(distances.keys())[:self.k]:
            neighbors.append(distances[key][-1])
        # return most common vote
        return max(set(neighbors), key=neighbors.count)
```

Figure 8.1: System Classification Code (a)

[13]. Here the k differs for the variety of data sets which is why k-value of several subjects are done. K=1 is regarded as the ideal point which leads to accuracy where it has gained optimal categorization. Not only this, a quality deviation value is being prepared by its reliable result. In KNN when k=1, it performs the best result by its features.

8.6 SVM

The EMG categorization accuracy rate invariably exceeds 90 percent by using an SVM. The only lingering and challenging task is the selection of configuration of the SVM and the selection of feature extraction from the EMG signal of the functionality. Although SVMs are binary classifiers, only two categories of gestures might be distinguished by a unified SVM. A multi-class SVM is expected to interpret multiple gestures, which can be conducted in one of two ways. Very first methodology could be to solve the interoperability problem that includes details from all multiple classes. The second alternative certainly postulates the multi-class SVM from a binary SVM confluence. The juxtaposition of binary SVMs is much simpler and does not impede the classification performance [6]. There are two plausible schemes using binary SVMs for enacting multi-class SVM which are: "one against all" (OAA) or "one against one" (OAO) [6]. OAA implements n binary classifiers to train each classifier to distinguish from the residual classes. OAO entails $(n(n-1))/2$ binary categories in which each category deviate from one pair of categories [6]. The penultimate classification in OAO is contingent on a voting mechanism, which comprises all the outputs of all classifiers. The ultimate outcome is the incident with the most votes. There is a more or less equivalent efficiency across both schemes as indicated in [6]. Nonetheless, the OAO scheme derives a stronger probability approximation across each class through evaluating all group pairs. This research incorporated with the EMG classification system with the OAO regime. We have first collected real data through EMG signal. Then we have done pre-processing and feature extraction of those signals using different methods. After that, we have classified our data set for getting the most accurate value for motion recognizing. Finally, we have feed those values to our simulated model and have tested the accuracy of our model which is represented in Table 8.7.

Table 8.7: Using support vector machine standard classification accuracy rate

Data Sub	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Avg. in %**
1st Sub	91	86	84	89	86	92	86	88	89	96	88.70
2nd Sub	89	94	85	96	79	97	89	87	88	97	90.10
3rd Sub	87	90	100	89	96	89	95	85	88	90	90.90
4th Sub	96	92	100	97	97	86	97	93	97	92	94.70
5th Sub	91	96	96	96	97	96	89	100	96	92	94.90
										Avg.	**91.86**

Here are some glimpse of the python code we applied in predicting the states at Fig 8.2.

```python
import matplotlib.pyplot as plt
from sklearn import datasets
from sklearn import svm
import numpy as np

from sklearn.externals import joblib

train_data = np.load('train_data.npy',encoding="lt1")

X = np.array([i[0] for i in train_data])
Y = [i[1] for i in train_data]
print(len(X))
print(len(Y))
print((X[0]))
print(Y[0])
print(Y)

#SVM
clf = svm.SVC(kernel='linear')
clf.fit(X,Y)
filename = 'dataset.sav'
joblib.dump(clf, filename)

clf1 = joblib.load(filename)
print("output")
print(clf1.predict([X[0]]))
print(clf1.predict([X[1]]))
print(clf1.predict([X[2]]))
print(clf1.predict([X[3]]))
#print(clf.predict([X[4]]))
# print(clf.predict([X[5]]))
```

Figure 8.2: System Classification Code(b)

For different hand gestures, we are predicting the possibilities of crossing the threshold which is determined by the algorithm itself. After the prediction, later on the value gets passed in the variables which is used for determining the state of hand simulation and giving command to the simulation after getting the online values from EMG sensor. After determining, the code sets the value of the variables for decided state by if else condition which sets commands the hand to be in different gestures, a glimpse of that code is shown below in Fig 8.3.

```
def emg_data():
    global ax
    if (state.get()):
     ax.cla()

    filename =  filedialog.askopenfilename(initialdir = "/",title =
    print(filename)
    file_str=str(filename)
    file_list=filename.split('/')
    print(file_list[-1])
    f_name=file_list[-1].split('.')
    ff_name=f_name[0]
    print(ff_name)
    file_name_read.set(ff_name)
with open(filename, 'r') as in_file:
     stripped = (line.strip() for line in in_file)
     lines = (line.split(",") for line in stripped if line)
     with open('0/'+ff_name+'.csv', 'w') as out_file:
         writer = csv.writer(out_file)
         writer.writerow(('title', 'intro'))
         writer.writerows(lines)
     return ff_name
clf = svm.SVC(kernel='linear')
   filename = 'EMG-model.sav'
   clf1 = joblib.load(filename)
   print("output")

   pred=clf1.predict([x])
   if(pred==0):
     v.set("state : HandRest")
     #print("normal")
   elif(pred==1):
     v.set("state : HandOpen")
     #print("normal")
```

Figure 8.3: System Classification Code(c)

8.7 Hardware Interfacing

To attain the value from the EMG sensor it was attached with an embedded micro-processor. The EMG sensor had 8 probes which were put on to different part of the arm. The probes got raw data from the arm and the sensor optimized it and sends it to the microprocessor. And the microprocessor sends that data for processing to the processing device. The embedded microprocessor which is used with EMG

```
int inByte = 0;
void setup() {
  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);
   establishContact();  // send a byte to establish contact until receiver responds
}

void loop() {
  // read the input on analog pin 0:
  int sensorValue = analogRead(A1);
  // Convert the analog reading (which goes from 0 - 1023) to a voltage (0 - 5V):
  float voltage = sensorValue * (5.0 / 1023.0);
  // print out the value you read:
  Serial.println(voltage);
  float a = voltage ;
  if (a > 0)
  {
    inByte = a;  // wait for a byte to be sent from the receiver
    int z_byte = a;  // read the value on analog input A0
    Serial.write(z_byte * 0.25);  // multiply the value by .25 and send it
  }
}
void establishContact()
{
  while (a<= 0)
  {
    Serial.print("No Sensor Value");   // send a capital A
    delay(300);
  }
```

Figure 8.4: EMG Interfacing Code

sensor is mainly bound by the analog pin 1. Before using the EMG sensor it was calibrated so that sensor can give accurate results.

Chapter 9

Result Analysis

9.1 Simulation

The EMG sensor has 8 electrode and the raw EMG data stand in for an 8-column matrix. Trial starting time and the values representing the label of each trial are declared by variables in the Autodesk file. After that the animation and python script was applied instead of any additional system for the simulation. To make sure our design could work well in real life, a simulation software named Autodesk 3D max is used in this work. 3D max is software for 3D modeling and rendering software for visualization, games, and animation. We did our simulated tests of 4 gestures accuracy in this software. We have shown our 3D models accuracy easily using 3D max's tool kits and motion techniques. However firstly, the 3D model of our prosthetic hand were made and then we have modified the model according to our work to meet the need for calculation and kept the most dimensions. We can get the test result by connecting the MyoWare Muscle Sensor with the embedded micro-processor by analogue pin. When the electrodes will collect muscle value from the subject the values pass through the processing unit by embedded micro-processor. The simulator checks if the variable from python script for deciding states. When collected value gets bigger comparing to threshold value, that certain variable sets its value to the right state. We have tested our gestures accuracy in the model and we have got accurate 4 gesture movement from our prosthetic model.

In Fig 9.1, when the online value from EMG sensor gets higher than the threshold

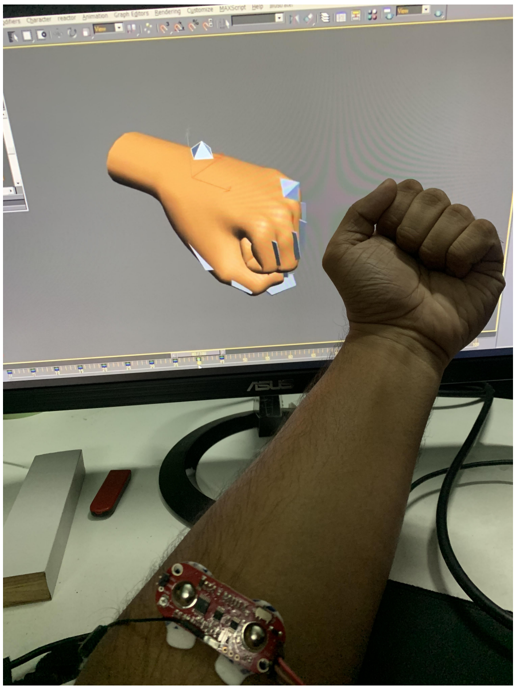


Figure 9.1: Hand-Rest Simulated view

Figure 9.2: Hand-Open Simulated view

Figure 9.3: Spherical-Grip Simulated view

Figure 9.4: Fine-Pinch Simulated view

value of “Hand-Rest” state, the variable determines the state by prediction from algorithm and sets the state in “Hand-Rest” posture of hand. As a result from the python script, Autodesk gets the command of the posture hand rest and represents it virtually through animation. For Fig 9.2, Fig 9.3 and Fig 9.4 the prediction, decision along with presentation is previewed in the simulation acquired in the same way.

9.2 Aperture Size vs. Classification Rate

Changing the aperture area, the divergence of classification rate is shown in Fig 9.5. Observing the aperture area change it is found that the expand of classification rate happens due to the increase of aperture with fixed aperture enhancement. 512 ms aperture area is the highest rate of classification rate. Substantial aperture takes substantial processing time and 512 ms aperture renders better classification percentage. In this study 128 ms as aperture size with fixed aperture enhancement has been chosen.

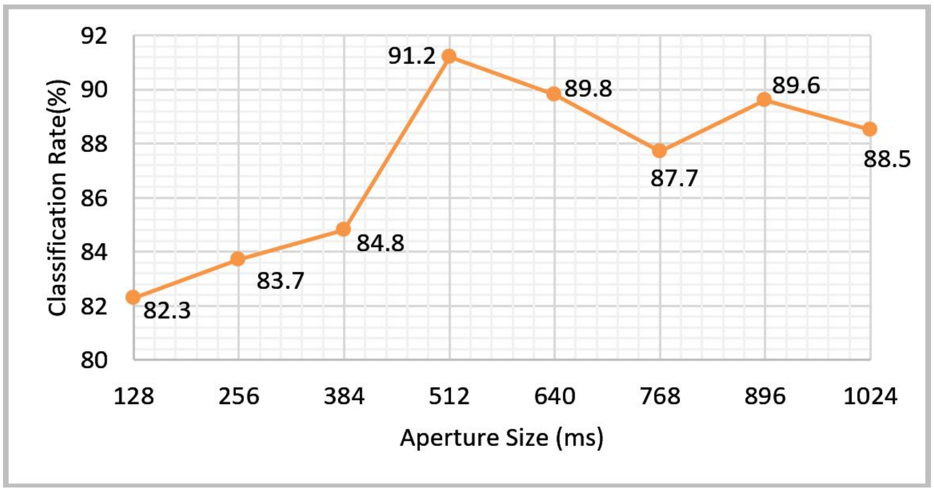


Figure 9.5: Relation between classification rate and aperture sizes

9.3 Characteristic Framework vs. Classification Percentage

Selection of characteristic frameworks assigning error rate or classification rate is shown on Fig 9.6. Standard error rate decreases due to the expanding characteristic parameters. Characteristic frameworks of characteristic set seven and eight from Table 8.1 manifests more fascinating outcome.

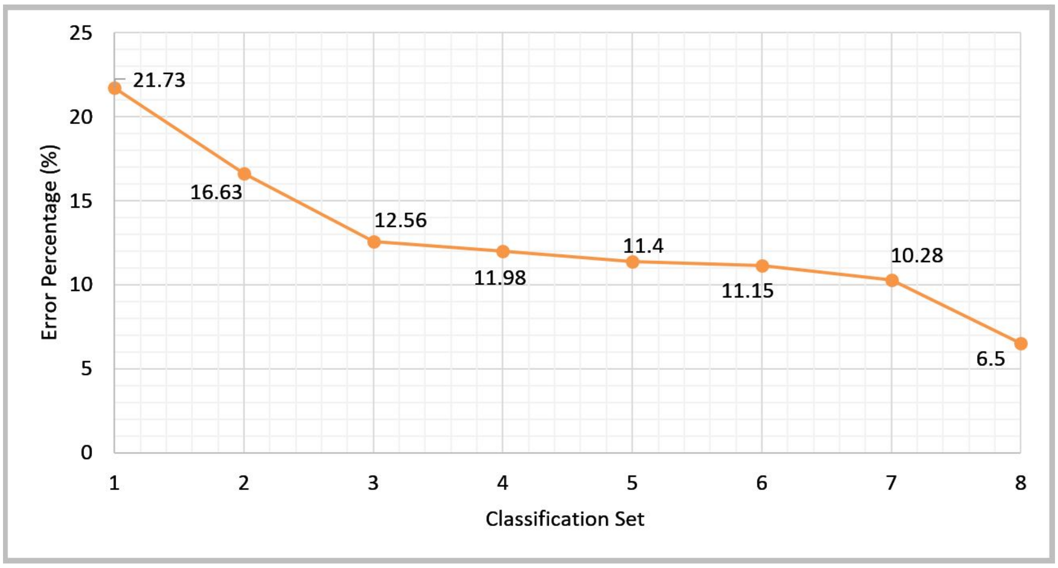


Figure 9.6: Error percentage and classification set relevance

According to categorization 7th and 8th combination, deduction of mean inaccuracy percentage is lesser. Error percentage does not lessen if classification set eight is used which also includes AR model. It has been found that classification set eight's classification parameter has been recon to systematize EMG patterns.

9.4 Average Categorization Rate

Comprehensive to All subject's average categorization rate is 96.33 percent. Fig 9.7 shows mean categorization rate for each subject where x-axis stand for specific subject and y-axis specifies each value according to the categorizing of each subject along with a horizontal line which represent overall mean categorizing rate.

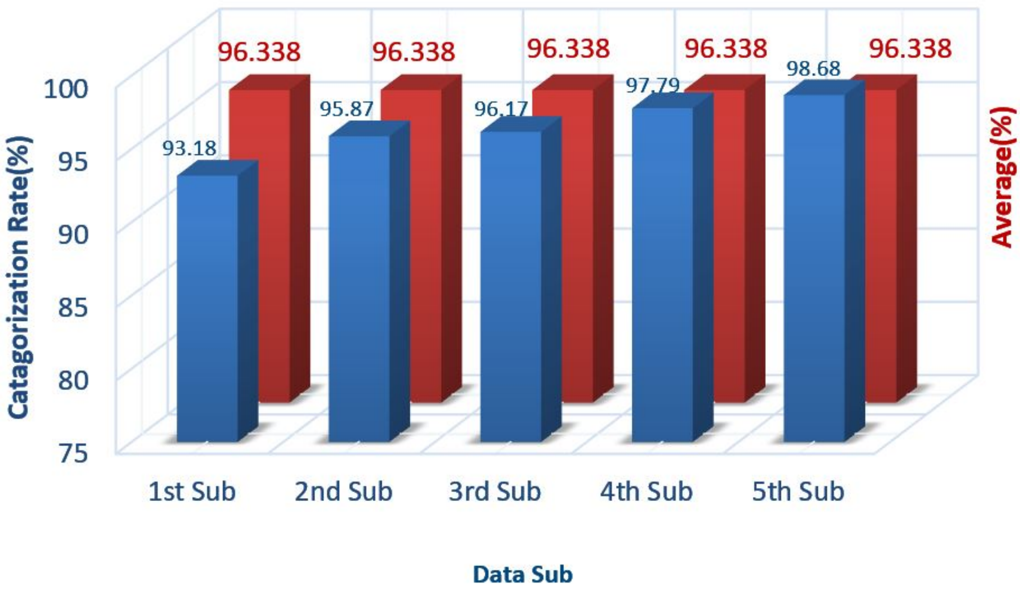


Figure 9.7: SVM-KNN classifier based mean categorization accuracy rate

Chapter 10

Limitations and Future works

10.1 Limitations

Though this system has an accuracy level close to 100 % and this is one of the precise predicting system, it has some limitations too. These are the limitations which were not possible to remove within the time binding we got and within the system budget we got. Some of the system limitations are stated bellow:

- Accuracy: The system has 96.33% of accuracy which concludes that the system is not 100% accurate and the inaccuracy rate is 3.67% which is certainly very low comparing other comparable system but it is not null.
- Feeling objects and temperature: The system has not described any solution about feeling objects, temperature and feeling texture of a surface just like the real organic hands.
- Gesture Variance: Unfortunately only four gestures were described throughout the whole project. Though classifying other gestures is very similar like classifying the four gestures.
- Rigidness about placements of electrodes: To place the electrodes on the muscles surface, the points has to be precise for every subject. It may work inaccurately in case of not placing the electrodes precisely on the designated point of muscle which is already described in the system documentation.
- Two algorithms: More algorithms could be used to experiment the accuracy variance by merging them together, though it is still a conceptual concept among the researchers.
- Degree of movement: The system has not accounted the fact of degree of movement. Degree of movement allows every move of hand or any parts of body with a variance of precise angle.
- Dataset: As we tried to keep the system budget friendly, we couldn't afford to buy more costly datasets for the system.

- No hardware implementation: The implementation is only shown through the simulation software, not through hardware implementation.
- Required hand portion: The system is valid for those people who have at least up to the elbow part of their hand. Unfortunately this is not valid for those amputees who lost their whole hand portion.

10.2 Future Works

The implemented prosthetic hand with 4 gesture movement control system is specific to biomedical applications which can be further extended. Recommendations for future works and major improvements is suggested below:

- Exploring another algorithms of Machine Learning which are related to the problem of hand gesture classification for comparing our existed results for experimenting the accuracy variance and merging them together if needed.
- Building a 3D printed hand and implementing our classifier as for now our implementation is shown only by software simulation system.
- Working on the points for placing the electrodes to be more precise for every subject.
- Increasing the accuracy to 100% for our classifier as now it has accuracy of 96.33%, so focusing more on the algorithms to get rid of inaccuracy of 3.67%.
- Replicating the results presented in this thesis in trials with actual amputees.
- Introducing gesture recognition system for both hand and also enabling the more natural interaction environments for amputees.
- Extending our results to other types of movements as in our thesis we have worked on 4 specific gestures only.
- Considering the Degree of Freedom in our classifier as this allows every move with a variance of angles.
- Introducing feedback system in our prosthetic hand so that amputees feel this hand as a part of their body by placing pressure sensors on the tip of the fingers.

In conclusion, we can say that although there are still much to do for the betterment of this system but the implemented classifier and the system is good and strong starting point for developing prosthetic hand systems.

Chapter 11

Conclusions

To conclude, we have used a unique approach named electromyography for our prosthetic hand which is controlled by the electrical signals instinctively produced by a person's muscle system. Using the Machine learning and deep learning the EMG based prosthetic hand movement control is designed, developed and represented in Autodesk 3D max software. At first, the system is implemented with different methods of extraction and classification of features. Three Algorithms are implemented using the methods of machine learning as well as classification accuracy is achieved in this work. For feature extraction, Variance, Waveform Length, Integral of EMG, Zero Crossings, Slope Sign Changes, Auto-regressive Model, Integrated Average Value and other techniques are used. In addition, for feature classification SVM, KNN, and combination of KNN and SVM algorithms are used.But the combination of SVM and KNN has the highest accuracy rate among other two algorithm as we get the accuracy rate of this classifier 96.33 percent while classifying the four different hand gestures which are hand-open, hand-close, spherical-grip, and fine-pinch. So, it can be easily concluded that the mix combination of KNN and SVM algorithm is more suitable for classifying different hand gestures with highest classification rate. We have also used the 3D modeling and rendering software name Autodesk 3D max for implementing our software simulation system. Output can be seen through the motion of our designed prosthetic hand model which performs accurately the intended action of the subject based on EMG signal test data. This model also takes muscles time for processing the motions according the data which is collected by MyoWare Muscle Sensor. In the near future, our 3D cost effective printable prosthetic hand with the accurate hand movement control system can be used by amputees to do the necessary activities in their daily regular lives to make their life easier.

Bibliography

[1] A. Prosthetics, *15 limb loss statistics that may surprise you*, https://accessprosthetics.com/category/living-with-limb-loss/, Access Prosthetics Blog, Living with Limb Loss, Oct 18, 2017.

[2] K. Kuribayashi, S. Shimizu, A. Kawachi, and T. Taniguchi, "A discrimination system using neural network for sma prosthesis," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'94)*, vol. 3, 1994, 1832–1839 vol.3.

[3] D. R. Merletti and P. di Torino, "Standards for reporting emg data," *Journal of Electromyography and Kinesiology JEK*, 1999.

[4] R. Li, S.-W. Ye, and Z.-Z. Shi, "Svm-knn classifier - a new method of improving the accuracy of svm classifier," vol. 30, pp. 745–748, May 2002.

[5] C.-W. Hsu, C.-C. Chang, and C.-J. Lin, "A practical guide to support vector classification," Department of Computer Science, National Taiwan University, Tech. Rep., 2003. [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/papers.html.

[6] M. Oskoei and H. Hu, "Oskoei, m.a.: Support vector machine-based classification scheme for myoelectric control applied to upper limb. ieee transactions on biomedical engineering 55(8), 1956-1965," *IEEE transactions on bio-medical engineering*, vol. 55, pp. 1956–65, Aug. 2008. DOI: 10.1109/TBME.2008.919734.

[7] H. S. Ryait, A. S. Arora, and R. Agarwal, "Study of issues in the development of surface emg controlled human hand," *Journal of Materials Science: Materials in Medicine*, 25 June 2008. DOI: https://doi.org/10.1007/s10856-008-3492-4.

[8] C.-C. Chang, C.-J. Lin, and C.-W. Hsu, *A practical guide to support vector classification*, http://www.csie.ntu.edu.tw/~cjlin/libsvm, A Library for Support Vector Machines, 2009.

[9] NettaGurari, K. Smith, M. Madhav, and A. M. Okamura, "Environment discrimination with vibration feedback to the foot, arm, and fingertip," *IEEE*, Jan. 2009.

[10] N. S. Rekhi, H. Singh, A. S. Arora, and A. K. Rekhi, "Analysis of emg signal using wavelet coefficients for upper limb function," in *2009 2nd IEEE International Conference on Computer Science and Information Technology*, 2009, pp. 357–361.

[11] M. Z. Al-Faiz, A. A. Ali, and A. H. Miry, "A k-nearest neighbor based algorithm for human arm movements recognition using emg signals," in *2010 1st International Conference on Energy, Power and Control (EPC-IQ)*, 2010, pp. 159–167.

[12] R. B. Gillespie, J. L. Contreras-Vidal, P. A. Shewokis, M. K. OMalley, J. D. Brown, HarshavardhanAgashe, RodolpheGentili, and A. Davis, "Toward improved sensorimotor integration and learningusing upper-limb prosthetic devices," Aug. 2010.

[13] K. KS, C. HH, M. CS, and M. CW, "Comparison of k-nearest neighbor, quadratic discriminant and linear discriminant analysis in classification of electromyogram signals based on the wrist-motion directions," vol. 11, pp. 740–745, 3 May 2011. DOI: \url{https://doi.org/10.1016/j.cap.2010.11.051}.

[14] J. L. G. Nielsen, S. Holmgaard, N. Jiang, K. B. Englehart, D. Farina, and P. A. Parker, "Simultaneous and proportional force estimation for multifunction myoelectric prostheses using mirrored bilateral training," *IEEE Trans. Biomed. Engineering*, vol. 58, no. 3, pp. 681–688, 2011. DOI: 10.1109/TBME.2010.2068298. [Online]. Available: https://doi.org/10.1109/TBME.2010.2068298.

[15] L. R., *Precise Finger Movements in Monkeys*, ser. Neuromethods. London UK: Humana Press, 2011, vol. 62. DOI: https://doi.org/10.1007/978-1-61779-301-1_22.

[16] J. Carpaneto, K. Somerlik, T. Krueger, T. Stieglitz, and S. Micera, "Natural muscular recruitment during reaching tasks to control hand prostheses," Jun. 2012, pp. 165–168, ISBN: 978-1-4577-1199-2. DOI: 10.1109/BioRob.2012.6290769.

[17] A. B. M. S. U. Doulah, S. A. Fattah, W. .-.-. Zhu, and M. O. Ahmad, "Wavelet domain feature extraction scheme based on dominant motor unit action potential of emg signal for neuromuscular disease classification," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 8, no. 2, pp. 155–164, 2014.

[18] J. Fajardo, A. Lemus, and E. Rohmer, "Galileo bionic hand: Semg activated approaches for a multifunction upper-limb prosthetic," in *2015 IEEE Thirty Fifth Central American and Panama Convention (CONCAPAN XXXV)*, 2015, pp. 1–6.

[19] *Amputee statistics you ought to know*, Limb Loss in the USA, 2016.

[20] T. Airsoft, *Electromyography (emg) sensors and signal processing*, https://www.youtube.com/watch?v=W5ZOg1x49Rc, Accessed 1/8/2019, 2017.

[21] J. Fajardo, V. Ferman, A. Lemus, and E. Rohmer, "An affordable open-source multifunctional upper-limb prosthesis with intrinsic actuation," in *2017 IEEE Workshop on Advanced Robotics and its Social Impacts (ARSO)*, 2017, pp. 1–6.

[22] A. Islam and M. Alam, "Classification of electromyography signals using support vector machine," Jan. 2017.

[23] S. C. B. Jaganathan, C. Bose, A. Badawy, J. Baili, B. Suresh, S. Changalasetty, W. Ghribi, and H. Bangali, "A hybrid ga/knn/svm algorithm for classification of data," Feb. 2019.

[24] H. Maamri, "Classifying hand gestures for prosthetic arms by identifying emg signals with machine learning and deep learning.," PhD thesis, Jun. 2019.

[25] S. Oyama, *Emg processing*, https://www.youtube.com/watch?v=6y6QDPBRaCc, Accessed 9/5/2019.

[26] K. Yashuk, I. Perikov, and M. Bulgakov, *Classify gestures by reading muscle activity*, https://www.kaggle.com/kyr7plus/emg-4/discussion.