TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

THAPATHALI CAMPUS

A Major Project Report

On

Human-Computer Interaction using Neuromuscular Signals

Submitted By:

Rabin Nepal (073/BEX/331)

Rhimesh Lwagun (073/BEX/333)

Sanjay Rijal (073/BEX/342)

Upendra Subedi (073/BEX/347)

Submitted To:

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

THAPATHALI CAMPUS

KATHMANDU, NEPAL

February 2021

TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

THAPATHALI CAMPUS

A Major Project Report

On

Human-Computer Interaction using Neuromuscular Signals

Submitted By:

Rabin Nepal (073/BEX/331)

Rhimesh Lwagun (073/BEX/333)

Sanjay Rijal (073/BEX/342)

Upendra Subedi (073/BEX/347)

Submitted To:

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

THAPATHALI CAMPUS

KATHMANDU, NEPAL

In partial fulfillment for the award of the Bachelor's Degree in Electronics and Communication Engineering

Under the Supervision of

Mr. Dinesh Baniya Kshatri

February 2021

DECLARATION

We hereby declare that the report of the project entitled **"Human-Computer Interaction using Neuromuscular Signal"** which is being submitted to the **Department of Electronics and Computer Engineering, IOE, Thapathali Campus**, in the partial fulfillment of the requirements for the award of the Degree of Bachelor of Engineering in **Electronics and Communication Engineering**, is a bonafide report of the work carried out by us. The materials contained in this report have not been submitted to any University or Institution for the award of any degree and we are the only author of this complete work and no sources other than the listed here have been used in this work.

Rabin Nepal (Class Roll No: 073/BEX/331) _____

Rhimesh Lwagun (Class Roll No: 073/BEX/333) _____

Sanjay Rijal (Class Roll No: 073/BEX/342) _____

Upendra Subedi (Class Roll No: 073/BEX/347) _____

Date: February, 2021

CERTIFICATE OF APPROVAL

The undersigned certify that they have read and recommended to the **Department of Electronics and Computer Engineering, IOE, Thapathali Campus**, a major project work entitled "**Human-Computer Interaction using Neuromuscular Signal**" submitted by **Rabin Nepal, Rhimesh Lwagun, Sanjay Rijal** and **Upendra Subedi** in partial fulfillment for the award of Bachelor's Degree in Electronics and Communication Engineering. The Project was carried out under special supervision and within the time frame prescribed by the syllabus.

We found the students to be hardworking, skilled and ready to undertake any related work to their field of study and hence we recommend the award of partial fulfillment of Bachelor's degree of Electronics and Communication Engineering.

Project Supervisor
Er. Dinesh Baniya Kshatri
Department of Electronics and Computer Engineering, Thapathali Campus

---

External Examiner
Dr. Bishesh Khanal
Director/Research Scientist at NAAMI

Project Coordinator
Er. Dinesh Baniya Kshatri
Department of Electronics and Computer Engineering, Thapathali Campus

---

Head of Department
Er. Kiran Chandra Dahal
Department of Electronics and Computer Engineering, Thapathali Campus

February, 2021

COPYRIGHT

ACKNOWLEDGEMENT

We would like to use this opportunity to express our deepest gratitude to our project supervisor, Mr. Dinesh Baniya Kshatri, for all the guidance and motivation he has provided. We are thankful to OpenBCI for sponsoring the major part of this project. We would also like to expand our gratitude to our lecturers who have helped us directly and indirectly.

Finally, we are very thankful to the Department of Electronics and Computer Engineering, IOE Thapathali Campus for providing us this opportunity to conduct this project.

Rabin Nepal (073/BEX/331)
Rhimesh Lwagun (073/BEX/333)
Sanjay Rijal (073/BEX/342)
Upendra Subedi (073/BEX/347)

ABSTRACT

Subvocal speech or internal articulation is a form of non-voiced speech that is voluntarily spoken. It is generated alongside the micromovement of the articulatory muscles that is imperceptible to others. However, the faint sEMG (surface Electromyography) signals can still be detected and analyzed to predict the internally articulated speech. This research project attempts to study this very phenomenon and its possible use case in human computer interaction. By extracting sEMG signals from several of these articulators and processing the extracted signals, prominent features of a particular utterance can be isolated and can be used to train a machine learning model. After training the model on several such utterances, accurate predictions of the utterances can be made which can be further utilized to perform a predefined action on a remote computer. This research also explores on improving traditional speech recognition models by possible augmentation of both approaches.

Keywords: Human computer interaction, Internal articulation, Speech recognition, Surface electromyography

TABLE OF CONTENTS

List of Tables

List of Figures

List of Abbreviations

ADC	Analog to Digital Converter
ALS	Amyotrophic Lateral Sclerosis
ANN	Artificial Neural Network
AR	Auto regression
ATP	Adenosine Tri Phosphate
AUC	Area Under Curve
BCI	Brain Computer Interaction
CMRR	Common Mode Rejection Ratio
CN	Cranial Nerve
CNN	Convolutional Neural Network
CSV	Comma Separated Value
CTC	Connectionist Temporal Classification
DAC	Digital to Analog Converter
DFT	Discrete Fourier Transform
DNPA	Double Nine Point Average
DTCWT	Dual Tree Complex Wavelet Transform
ECG	Electrocardiography
ECoG	Electrocorticography
EDA	Electronic Design Automation
EEG	Electroencephalography
EMG	Electromyography
EMS	Electrical Muscle Stimulation
EoG	Electrooculography
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
GUI	Graphical User Interface
HFS	High Frequency Signal
HPF	High Pass Filter
ICA	Independent Component Analysis
IDE	Integrated Development Environment

IIR	Infinite Impulse Response
IoT	Internet of Things
KNN	K-Nearest Neighbor
LPF	Low Pass Filter
LSB	Lower Significant Bit
MLP	Multi-Layer Perceptron
MUAP	Motor Unit Action Potentials
NEMS	Neuromuscular Electrical Stimulation
PCA	Principle Component Analysis
PCB	Printed Circuit Board
PGA	Programmable Gain Amplifier
PNS	Peripheral Nervous System
RCA	Radio Corporation of America
ReLU	Rectified Linear Unit
RHFS	Rectified High Frequency Signal
RNN	Recurrent Neural Networks
ROC	Receiver Operating Characteristics
SNR	Signal to Noise Ratio
SPI	Serial Peripheral Interconnect
STFT	Short-Time Fourier Transform
TTL	Transistor Transistor Logic
UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus
WER	Word Error Rate
ZCR	Zero Crossing Rate

1. INTRODUCTION

Communication is an integral part of humans. Communication enables humans to exchange information and messages. There are different means of communication but the very intriguing and primitive form of communication is speech.

1.1 Background

Speech is human vocal communication, where information is exchanged with different sounds produced. This is a very simple method and is implemented by many of the modern gadgets taking it as an input from the user. This kind of interaction between a human and a computing device has existed for decades but such interface has recently been reliable and fast enough to be used in real world applications. The more traditional approach for human computer interface is peripheral devices like mouse, keyboard, touchpad, stylus, joysticks, etc. Despite the advancement in verbal human computer interface, such traditional approaches are yet to be replaced.

Nevertheless, the advancement of neuroscience has introduced new methods of interacting with a computer. These kinds of interfaces are termed as brain computer interface. The input signal for the computer in such cases are extracted from the brain through process such as EEG. Operating a brain-controlled device then becomes as simple as thinking about controlling the device. This interaction method, however, does come with some major hindrances. The user must train themselves to concentrate on doing a specific task and not wander along with thoughts. There is also an ethical dilemma associated with such interface which questions if peeking inside someone's brain and their thoughts should be allowed. Another major drawback of such interface is the large number of input channels that needs extensive technical analysis to interpret any useful information.

An intermediate method between voiced interface and brain-controlled interface could be "Silent Speech Interface". Unlike voiced interface, this method is completely silent and unlike brain computer interface, this method is voluntary; a user has to deliberately

convey information through unvoiced form of speech. Silent Speech Interface is more seamless than voiced interface and requires no special user training at all. It utilizes the sEMG signals extracted from articulatory muscles (during voluntary speech production) using special electrodes and uses these signals to predict what the user is speaking internally. Besides sEMG there also exist many other techniques for studying internal speech articulation. They can be listed as follows:

- Electroencephalography (EEG)
- Electrocorticography (ECoG)
- Permanent Magnet Sensors
- Ultrasound
- Optical Camera
- Vocal Tract Resonant Signals

1.2 Motivation

Motivation is one of the factors, which encourages a person to the commitment of any action and play a crucial role as psychological determinant. There are many researches and projects which have been motivating us in different stages of our project. One of the major sources of motivations was AlterEgo.

AlterEgo is a personalized wearable silent speech interface that allows its users to silently converse with a computing device without any voice or any discernible movements - thereby enabling the user to communicate with devices, AI assistants, applications or other people in a silent, concealed and seamless manner. A user's intention to speak and internal speech is characterized by neuromuscular signals in internal speech articulators that are captured by the AlterEgo system to reconstruct this speech. This interface is used to facilitate a natural language user interface, where users can silently communicate in natural language and receive aural output (e.g.: - bone conduction headphones), thereby enabling a discreet, bi-directional interface with a computing device, and providing a seamless form of intelligence augmentation. [1]

This project is selected as it has interfacing of brain signals with computer. Brain Computer Interface (BCI) is new field for research and product development. In this context, this project provides platforms for research on this field. It is also intended to help on one of the evolving fields of science and technology i.e. bio-technology.

1.3 Problem Definition

Since the development of the very first computer, human-computer interaction has always required to have some form of physical action as an input to the computer. These traditional input devices include keyboards, mouse, joystick, cameras, microphones etc. Although these methods possess high accuracy and convenience, they suffer from lack of privacy. And it may not be possible for everyone to use such traditional means for interacting with a computing device. The traditional system has also been proved to be very slow in today's world. Speech interaction somewhat tackles this issue but is still subjected to privacy problems. The proposed system in this project tackles all these problems and provides a secure and faster interaction between a human and a computing device. And the proposed system works the same for everyone disregarding their disabilities.

1.4 Project Objectives

The main objectives of the project are:

- To extract and transfer EMG signals from articulatory muscles to a computer
- To process and convert the articulated speech signals to text

1.5 Project Scope

This project explores EMG signals generated in the speech articulatory muscles as a foundational element to human speech and tries to achieve a simple human computer interaction through it. This project does not include the decoding of signals produced during articulation of sentences and special characters. It is not compatible for languages other than English. The interface is unidirectional and does not include control of any device.

This project carries with it an extensive potential in regards to both aid to human potential and apprehension of the speech processes in humans. It possesses the ability to directly help speech impaired people and improve upon existing communication methods. Complex process of speech production can be simplified and studied further to understand hidden patterns and thus refine existing speech recognition models.

1.6 Project Applications

A. Silent Means of Communication

Internal articulation is inaudible mode of speaking that is only perceptible to the speaker themselves. Lack of any perceptible sound makes it perfect for any cohort applications where privacy is crucial. Similarly, this means of communication is also suitable in situations where voiced speech is not appropriate such as in libraries, meetings, etc.

B. Novel Human Computer Interface

Traditional human computer interface takes place through peripheral devices like mouse, keyboards, touchpads, etc. These traditional interfaces are often slow and inconvenient to people of all backgrounds, especially for people with disabilities. There exist voiced interfaces for differently abled peoples but those with speaking disabilities cannot utilize even this interface. Such is the case for people suffering from ALS (Amyotrophic Lateral Sclerosis) and other articulation disorders. Exploiting sEMG signals from speech articulators as an input to computing devices can help people suffering from such disabilities. In addition to this, controlling robotics limbs, IoT and other remote devices truly help it make a novel means of human computer interface.

C. Improvement of Speech Recognition Models

Current Speech Recognition models require huge amounts of data and tremendous computing power to obtain usable accuracy in recognition tasks. These models still cannot generalize well to different scenarios such as background noises, difference in speaking rate, difference in pronunciation, etc. sEMG signals from targeted articulatory muscles could provide new perspective to Speech Recognition tasks. These signals along with acoustic speech data could help improve the existing Speech Recognition models.

1.7 Report Organization

The material presented in this report is organized into eleven chapters. Chapter 1 is an introduction section which mainly describes the background, objective, scope and application of the project. It also focuses on the need of the project. Chapter 2 presents a brief summary of all existing works that has already been carried out in the related field. Chapter 3 provides information on generation and working of neuromuscular signals, explains the role of these signals in controlling the facial muscles during speech and how the internal articulation works along with the muscles involved. Chapter 4 illustrates the mathematical models required in the project and about the effect of noise and the type of noise that the designed system is susceptible to. Chapter 5 explains the implementation process of hardware components, software platforms and dataset used in the project. Chapter 6 elaborates a particular sequence in which the work has been carried out along with detailed procedures, block diagram or data flow diagram which illustrate the explanation of how the hardware and software were used to accomplish the project. Chapter 7 also contains the details of implementation of the procedures that have been explained in the methodology. Chapter 8 contains results of the project. The output is shown in graphical form as well. Chapter 9 contains analysis of overall project and its discussion. Chapter 10 contains possible future enhancement that can be done on this project. Chapter 11 concludes the overall project. Finally, chapter 12 contains the additional topics like project budget, project timeline, some plots and references used for the project.

2. LITERATURE REVIEW

There has been a number of attempts in electrophysiology for analysis of neural activities. “AlterEgo: A Personalized Wearable Silent Speech Interface”, a research accomplished by Arnav Kapur, Shreyas Kapur and Pattie Maes which was published by MIT Media Lab in 2018, presents a natural extension of the user's own cognition by enabling a silent, discreet and seamless conversation with machines and people. It presents a wearable silent speech interface that allows users to provide arbitrary text input to a computing device or other people using natural language, without discernible muscle movements and without any voice command i.e. without explicitly saying anything. The nerve impulses were sourced as seven channels from laryngeal region, hyoid region, levator anguli oris, orbicularis oris, platysma, anterior belly of the digastric mentum using electrodes on the outer skin [2].

According to neuroprosthetics experiment, “Control Machines with your Brain” done from 2009-2017 by Backyardbrains, a team of researchers and engineers, the EMG signals were extracted from Muscle SpikerShield which was interfaced with micro-controller and the data obtained was visualized in Spike Recorder App developed by the team. The research was further extended as Human-Machine Interfaces, which included control of robotic arm, video games and voiceless communication. [3]

A research paper by Michael Wand and Tanja Schultz named “SESSION-INDEPENDENT EMG-BASED SPEECH RECOGNITION” describes the method of speech recognition by surface EMG signals. By recording the electric active potentials of human articulatory muscles, it can be decoded into a speech that person is vocalizing. Speech recognition using EMG signals dates back to 1980s. 93% accuracy was observed on 10-word vocabulary. It suggests that good result can be obtained even for the signals taken when words are silently articulated. [4]

“Electrical Stimulated as a Modality to Improve Performance of the Neuromuscular System”, a research paper by Vanderthommen Marc and Duchateaus Jacques in October 2007 transcutaneous neuromuscular electrical stimulation (NEMS) can modify the order of motor unit recruitment and has a profound influence on the metabolic demand

associated with producing a given muscular force. Tetanic contractions elicited by pulses of high intensity and short duration induce a high metabolic stress in the muscle, contribute to the reversal of motor unit recruitment, and improve the maximal capability of the neuromuscular system primarily not only through increased force-generating capacity of the muscle but also through intensified voluntary activation. [5]

A research paper “End-to-end neural networks for subvocal speech recognition” written by Pol Rosello, Pamela Toman and Nipun Agarwala attempts to perform session independent subvocal speech recognition by leveraging character-level recurrent neural networks (RNNs) and the connectionist temporal classification loss (CTC). They utilized EMG-UKA trial coprus’s two hours of data to train their CTC models. Although the accuracy of their model is not mentioned, they did express some measures to improve the field to silent speech recognition through EMG signals in their paper. [6]

Munna Khan and Mosarrat Jahan wrote a paper “The Application of AR Coefficients and Burg Method in Sub-vocal EMG Pattern Recognition” which showcases successful recognition of Hindi phonemes (Ka, Kha, Ga, and Gha) with accuracy of about 75.5% to 80%. They studied burg algorithm techniques for EMG spectral analysis and used reflection coefficients and AR coefficients as features of sub-vocal EMG signal to recognize the patterns of sub-vocal phonemes. They concluded that the pattern recognition in EMG signal using reflection coefficients and AR coefficients is highly efficient and can be used to develop a real time module. [7]

In “Development of sEMG sensors and algorithms for silent speech recognition”, a research paper published by Geoffrey S. Meltzner, James T. Heaton et al., a new system capable of recognizing silently mouthed words and phrases based completely on surface EMG signals has been described. They tested a system of sensors and algorithms during a series of subvocal speech experiments involving more than 1,200 phrases generated from a 2,200-word vocabulary and obtained 91.1% recognition rate i.e. word error rate (WER) of 8.9%. They prepared their dataset performing experiments on a total of 19 subjects (11 males and 8 females) ranging from 20-42 years in age with no speech or hearing disabilities. They had applied discrete-cosine Fourier transform in order to obtain coefficients of the signal for the training set. [8]

Chuck Jorgensen and Kim Binsted in 2005 published a paper “Web Browser Control Using EMG Based Sub Vocal Speech Recognition” which describes they had trained six subvocally pronounced control words, 10 digits, 17 vowel phonemes and 23 consonant phonemes using a scaled conjugate gradient neural network. They had recorded the surface EMG signals of frequency range 30-500 Hz from the larynx and sublingual areas below the jaw, filtered them, sampled them at 2000 Hz and transformed into features using a Kingsbury’s Dual Tree complex wavelet transform (DTCWT) and short time Fourier transform (STFT). They had also designed a notch filter to eliminate line noise at 60 Hz. They had obtained an average of 92% accuracy. Using the trained control words, they performed sub vocal web browsing. [9]

Recent work by David Gaddy and Dan Klein titled as "Digital Voicing of Silent Speech" converts silently mouthed words to audible speech based on EMG sensor measurements. The EMG electrodes placed on neck and face region captures the EMG signal produced during silently mouthed words and generate synthetic speech. They have used zero crossing, high frequency, rectified high frequency, double nine point average and frame based power as the temporal features and STFT and MFCCs as spectral features. They used both vocalized EMG data and silent EMG data for feature alignments and train the model. The EMG feature of the instance pairs were aligned with dynamic time wrapping then refinements to the alignments were done using canonical correlation analysis. These aligned features were used as targets for training a recurrent neural transduction model. A WaveNet decoder was used to generate audio from predicted speech features. They have released dataset of EMG signal of both silent and vocalized speech. The dataset contains nearly 18 hours of facial EMG signals from a single speaker, with 9829 words in vocabulary. The dataset consist of parallel silent/vocalized speech and non-parallel vocalized speech. The parallel silent/vocalized speech contains 3.6 hours of silent speech and 3.9 hours of vocalized speech. The Non-parallel Vocalized speech contains 11.2 hours of recording. They compared their methods with other methods which shows their methods having 68% WER(Word Error Rate) and out-performs other method with a 20% absolute improvement. They concluded that it is still challenging for large set of vocabulary but also show promise as an achievable technology. [10]

3. ELCTROPHYSIOLOGY OF SPEECH PRODUCTION

Movement of body parts of living beings, voluntary as well as involuntary, is fully coordinated and controlled by the brain. Brain performs this controlling and coordinating activity through electrical signals.

3.1 Neuromuscular Signals

Electrophysiology is a branch of neuroscience that studies the electrical properties of biological cells and tissues. It also includes the measurements of electric current or voltage changes in biological cells [11]. Common types of electrical bio-signals are: EEG (Electroencephalogram), ERG (Electroretinogram), EMG (Electromyography), EOG (Electrooculography) and EGG (Electrogastrogram).

Normally, in biological cells the concentration gradient of potassium is greater from inside towards outside of the cell membrane so there is a strong tendency of extra K^+ ions to diffuse outward through the membrane. Diffusion of K^+ ions outside the cell membrane creates electro-positivity outside the membrane and electronegativity inside the membrane due to negative ions that remain behind and do not diffuse outward with the potassium ions. Within a millisecond the potential difference between the inside and outside of the cell membrane, called the resting or diffusion membrane potential becomes high enough to resist the further K^+ ions diffusion. In normal human nerve fiber, the resting potential is about -90 mV. The membrane is said to be polarized at this stage. Membrane then suddenly becomes very permeable to the Na^+ ions allowing a large number of Na^+ ions to diffuse to the interior of the membrane. Again, the membrane potential rises high enough within milliseconds and blocks further diffusion of sodium ions inside the membrane. The normal resting potential, -90 mV is neutralized; this is known as depolarization of the membrane. In large nerve fibers excess of Na^+ ions diffusion inside cause membrane potential to overshoot beyond the neutral point which is known as action potential. Within a $1/10000^{th}$ of a second, sodium channels begin to close and the potassium channels open more than normal. There is continuous pumping of three Na^+ ions outside the membrane for each two K^+ ions pumped inside. Rapid

diffusion of K^+ ions re-establishes the normal resting potential. This process is called repolarization. [12]

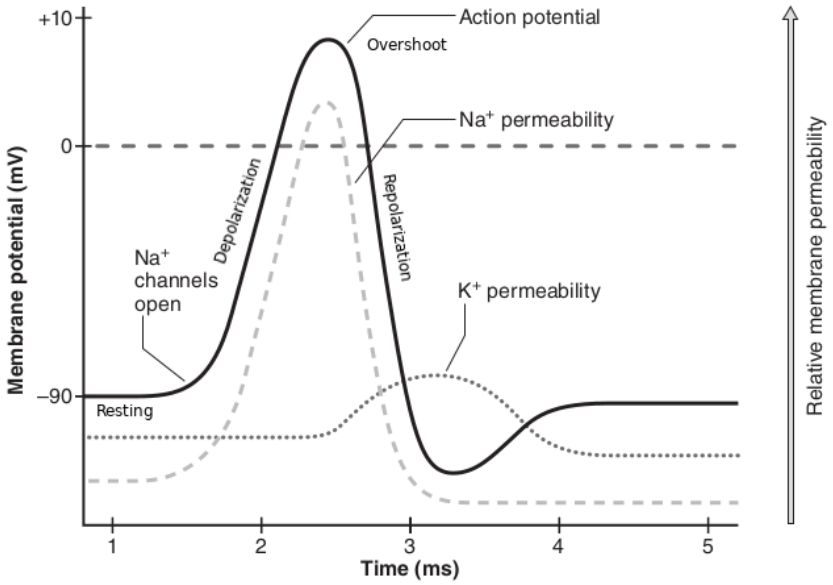

Figure 3-1: Time Course of the Muscle Fiber Action Potential

At neuromuscular junctions, the nerve fibers form a complex of branching nerve terminals that invaginates into the vicinity of the muscle fiber. In the axon terminal are many mitochondria that supplies Adenosine Triphosphate (ATP) for the synthesis of an excitatory neurotransmitter, 'acetylcholine' which excites the muscle fiber membrane. The neuromuscular junction (skeletal muscle fiber) is as shown in figure below.

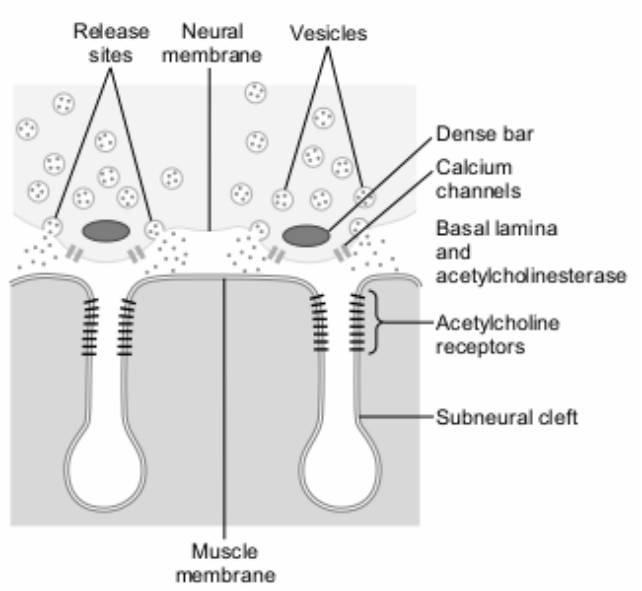

Figure 3-2: Neuromuscular Junction

When a nerve impulse reaches the neuromuscular junction, about 125 vesicles of acetylcholine are released from the terminals into the synaptic space and then continue to activate the acetylcholine receptors as long as it persists in the synaptic space, which is at most a few milliseconds. The elicitation of the acetylcholine receptors opens the acetylcholine gated channels at the muscle fiber membrane. The principal effect of opening these channels is to allow the large number of Na^+ ions to diffuse inside the fiber, carrying with them a large number of positive charges. This induces electrical potential inside the fiber at the local area of the end plate which is called the end plate potential (50-75 mV). The end plate potentials if not weakened by the toxins are strong enough to initiate the action potential as shown in figure 3-3. The end plate potential A and C are recorded from the muscles weakened by toxins; curare and botulinum respectively. The action potential then spreads along the muscle fiber membrane. After the generation of action potential, there is a brief refractory period during which membrane can't be stimulated, this prevents the message from being transmitted backward. [12]

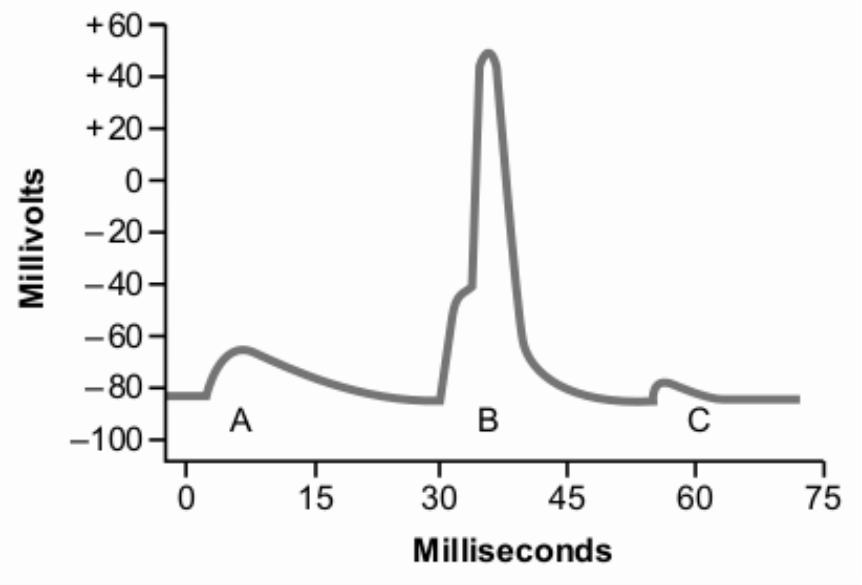

Figure 3-3: End Plate Potentials A, B and C with Action Potential at B

The skeletal muscle fiber is so large that the action potential spreading along its surface membrane causes almost no current flow in the fiber. Yet, to cause maximum muscle contraction, current must penetrate deeply into the muscle fiber to the vicinity of the separate myofibrils. This becomes possible due to transmission of action potential along the transverse tubules (T-tubules) that penetrate deep all the way through the muscle fiber.

The T-tubules action potential results in opening of the voltage gated calcium channels located at each side of the dense bar as shown in figure 1-2. The Ca^{++} ions then diffuse from the synaptic space inside the muscle fiber in the immediate vicinity of the myofibrils and cause the muscle contraction. This overall process is called excitation-contraction coupling. The signal thus generated on the muscle fibers is known as Electromyography (EMG) signals which can be measured using electromyography. [12]

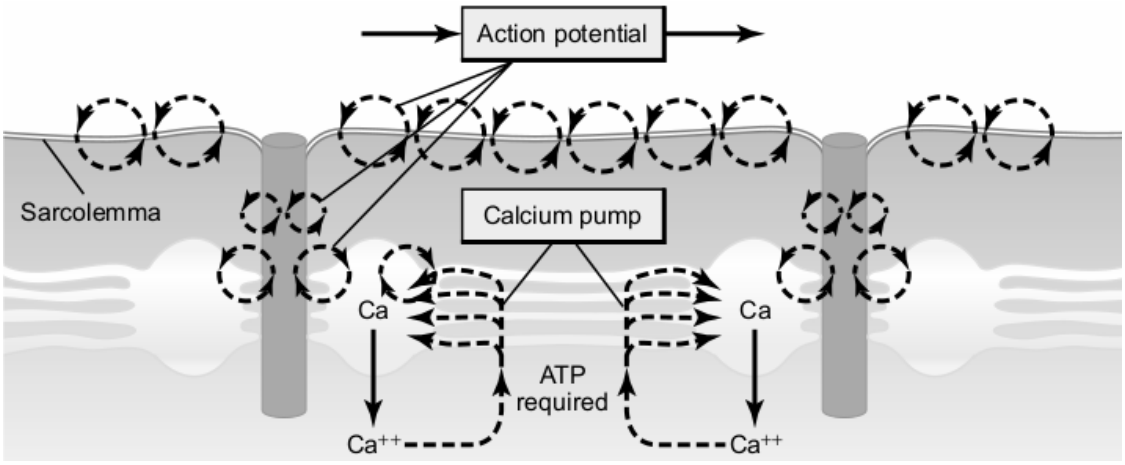

Figure 3-4: Excitation-Contraction Coupling in the Muscle

The Ca^{++} ions exert attractive force on the acetylcholine vesicles drawing them to the neural membrane. These vesicles then fuse with the neural membrane and empty their acetylcholine into the synaptic space by the process of exocytosis. During this process acetylcholine is rapidly removed by the acetylcholinesterase which decomposes acetylcholine into acetate and choline. The short time during which acetylcholine remains in the synaptic space (few milliseconds) is enough to excite the muscle fiber. The rapid removal of acetylcholine prevents continued muscle contraction. The number of vesicles present in the nerve endings is sufficient to allow transmission of only a few thousand nerve-to-muscle impulses. So, for continued transmission neuromuscular signals, new vesicles need to be reformed rapidly. Within a few seconds after each action potential is over, coated-pits appear in the terminal membrane caused by the contractile proteins in the nerve ending. Within a few seconds the protein contracts and causes the pits to break away into the interior of the membrane thus forming new vesicles. Extracellular fluids (ECF) reuptake choline from decomposed acetylcholine which then combines with the

free chlorine, acetyl coenzyme A, ATP and glucose to form acetylcholine in cytoplasm and then transferred to the vesicles within another few seconds. [13]

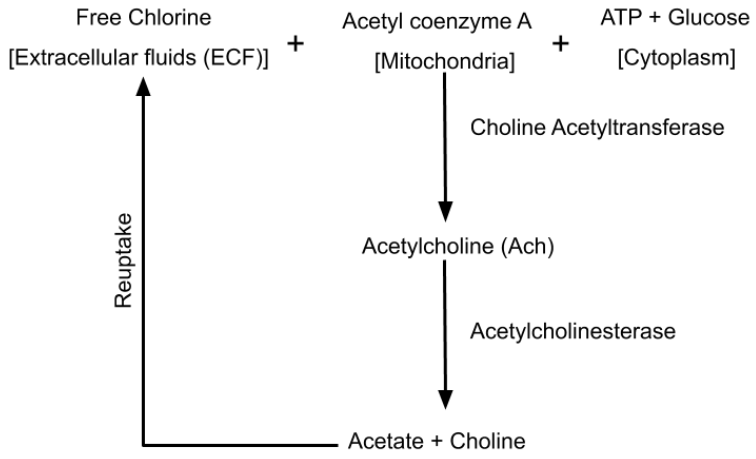

Figure 3-5: Formation and Decomposition of Acetylcholine (Neurotransmitter)

For the analysis of superimposed motor unit action potentials (MUPAs) generated from several motor units, the EMG signals can be decomposed into their constituents MUPAs distinguished by their characteristic shapes. The shape and size depend on where the electrode is located with respect to the fibers and are different if the electrode position is altered slightly.

3.2 Speech Production Mechanism

Human speech is a very complex process. It can be exemplified using a three-stage model of Conceptualization, Formulation and Articulation. The first two stages occur in the brain itself whereas the last stage occurs in the motor system under the control of the brain. Articulation again can be subdivided into three stages: Respiration, Phonation and Articulation.

3.2.1 Respiration

Respiration, also known as breathing, refers to the inhalation and exhalation of air by the contraction and expansion of diaphragm. During inhalation, the lungs expand, causing the air to flow from the mouth to the lungs with the glottis relatively open. During exhalation, the lungs contract, pushing the air from the lungs toward the mouth, which provides energy for human speech. The energy is given as a stream of air coming from

the lungs which passes through trachea and the vocal fold as shown in figure 3-6, where the phonation occurs. For most languages, the production of sound occurs during exhalation which is why humans cannot generally speak while inhaling.

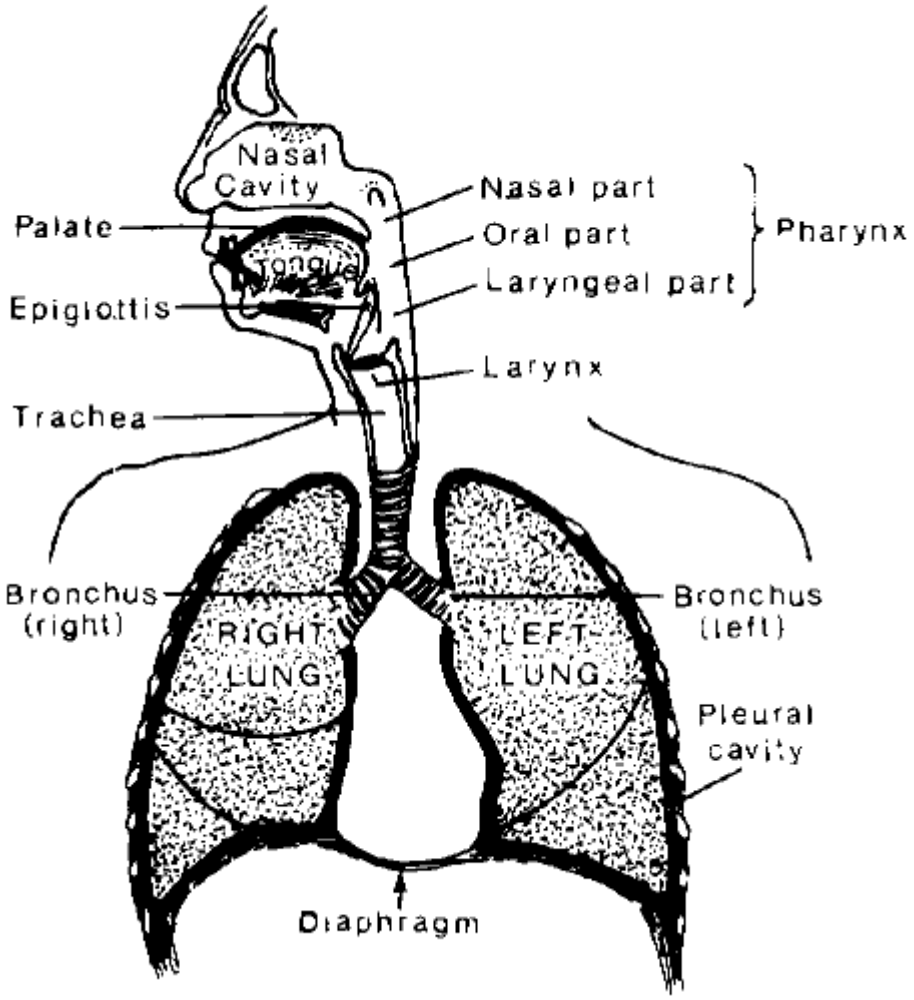

Figure 3-6: Vocal Apparatus in Humans

3.2.2 Phonation

Phonation is the process that modifies the pulmonic air in such a way that it produces acoustic signals. During phonation, the vocal folds vibrate causing a change in air pressure generating acoustic waves which then gets amplified internally through resonance. The voice source is also used to change the sentence melody and the tonal form of words by varying the subglottal pressure as well as the tension of the vocal folds.

This leads to changes in the rate of vibration of the vocal folds, which are in turn perceived by the listener as modifications in pitch and/or in loudness. [14]

3.2.3 Articulation

Articulation is the term used for all actions of the organs of the vocal tract that affect modifications of the signal generated by the voice source. This modification results in speech events which can be identified as vowels, consonants or other phonological units of a language [14]. The air stream is manipulated by several mobile organs called active articulators. The major active articulators are lower lip, tongue, glottis and uvula. A number of passive articulators such as: palate, nasal cavity, epiglottis, lower teeth, alveolar ridge, etc. supports the active articulators.

Table 3-1: Nerves and Muscle Movements in the Articulatory System

S.N.	Nerve	Movements	Sensory Functions
1	Trigeminal Nerve (CN V)	Biting and chewing	Sensory data from palate, teeth and anterior tongue
2	Facial Nerve (CN VII)	Facial muscle	Sensation to the external ear
3	Glossopharyngeal Nerve (CN IX)	Elevation of Pharynx and larynx	Sensation to posterior tongue and upper pharynx
4	Vagus Nerve (CN X)	Elevation of the palate phonation	Sensory data from external ear, tongue and larynx
5	Hypoglossal Nerve (CN XI)	Movement of the tongue	Sensory data from the tongue

During the production of speech, a complex series of finite and coordinated neuromuscular communication is associated. For the complete generation of sound more than 100 muscles are involved. When a person articulates a word internally without

acoustic vocalization and no significant movements in facial muscle and tongue, more than 15 muscles which are parts of the speech system, are neurologically activated. These particular muscles receive feeble electrical signals from the PNS. Nerves involved in the articulatory system and respective muscles movement are tabulated below.

3.2.3.1 Articulatory Muscles

Different facial muscles are involved in the activation of the active articulators for articulation. They are divided into different regions which are listed below:

- Labial region
- Lingual region
- Mandibular region
- Palatal region
- Pharyngeal region

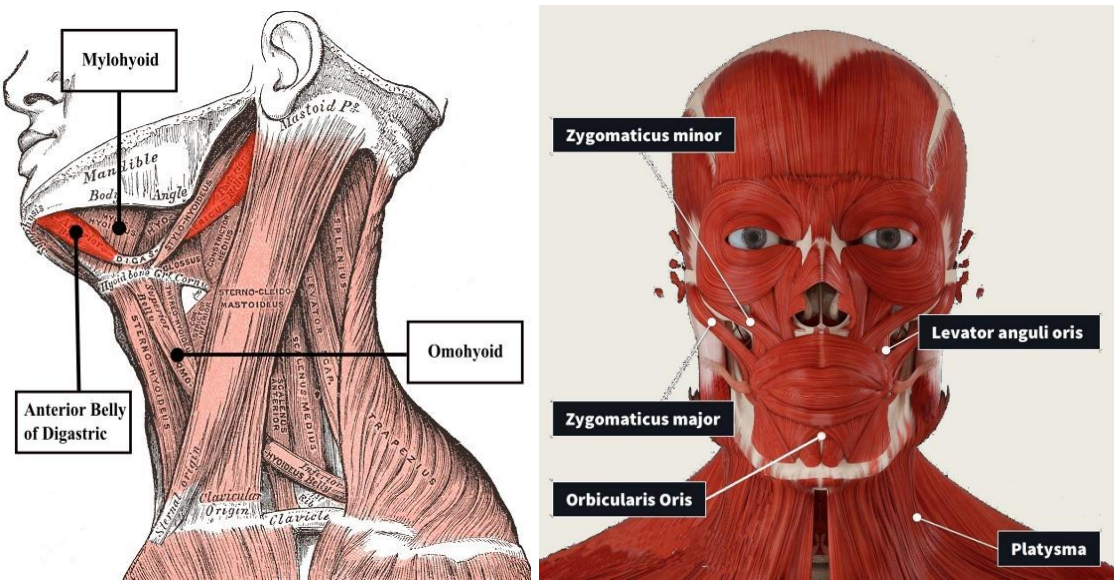

Figure 3-7: Active Facial Muscles

Selecting the right muscle for extracting the EMG signal is very essential. Many factors such as noise susceptibility, signal strength, cross-talk, signal frequency, etc. depend on the selection of the muscle. Some of the basic criteria to be followed while selecting the muscles are as follows:

- Select the muscle where it is convenient to place the electrodes.
- Distance from the electrode to the muscle should be minimum.
- Size of muscle should be large enough to avoid cross-talks.
- Configuration to be followed for signal extraction (bipolar or monopolar) also determines which muscle to select.
- Size of the electrode and the type of electrode should be considered before selecting the muscles.

For 8 channel hardware setup, different muscles in the facial regions are selected along with a reference and a ground. Followings are the muscles from where EMG signals were extracted.

A. Levator Anguli Oris

Levator Anguli Oris is a facial muscle close to the mouth opening that lifts the angle of the mouth. This muscle is innervated by the buccal branch of the facial nerve (CN VII). When activated, the levator anguli oris lifts the angle of the mouth, thus participating in creating a smile. Contractions of this muscle produce a facial expression associated with self-confidence.

B. Zygomaticus Minor

The Zygomaticus minor is the muscle responsible for facial expressions. It draws the upper lip backward, upward and outward while smiling. It originates from zygomatic bone and continues with orbicularis oculi on the lateral face of the levator labii superioris and then inserts into the outer part of the upper lip. It is innervated by the facial nerve (CN VII).

C. Zygomaticus Major

The zygomaticus major muscle is a paired facial muscle that extends between the zygomatic bone and the corner of the mouth. It is one of the two zygomatic muscles (major and minor) that lie next to each other in the cheek area. An activated zygomaticus major muscle is involved during smile. The nerve supply of the zygomaticus major is received from the zygomatic and buccal branches of the facial nerve (CN VII).

D. Orbicularis Oris

Orbicularis oris muscle, also known as musculus orbicularis oris is a complex, multi-layered muscle which encircles the mouth and plays a role in facial expression. It is an attachment site for many other facial muscles around the oral region. It controls the movement of mouth and puckering of the lips.

E. Omohyoid

Omohyoid muscle is located at the anterior part of the neck. It consists of two bellies separated by an intermediate tendon. The function of this muscle is to depress the larynx and re-establish breathing following the act of swallowing and speech production.

F. Anterior Belly of Digastric

The anterior belly of the digastric (Latin: venter anterior musculo digastrico) is one of the two bellies of the digastric muscle. The anterior belly is smaller than the posterior belly, and it develops from the first pharyngeal arch. Upon contraction the anterior belly of the digastric muscle elevates the hyoid bone. The anterior belly of the digastric is innervated by the mylohyoid nerve, which arises from the mandibular division of the trigeminal nerve (CN V3).

G. Mylohyoid

The Mylohyoid muscle is one of the muscles that forms the floor of the oral cavity. It is responsible for the control of the tongue that makes velar consonants and vowels. Velar consonants are those consonants that are produced from the back of the tongue. The main function of this muscle is to facilitate the speech. Mylohyoid muscle is located in the middle of the neck under the chin.

H. Platysma

The platysma (also platysma muscle, Latin: platysma) is a wide, flat, superficial neck muscle extending from the lower part of the face to the upper thorax. The platysma is a paired thin and superficial muscle arising from the upper parts of the shoulders and inserting into the mouth area. Contractions of the platysma depress and wrinkle skin of the lower face and the mouth. The platysma also contributes to forced depression of the mandible. The platysma is innervated by the cervical branch of the facial nerve (CN VII).

3.3 Internal Articulation

Internal articulation is a significantly attenuated form of speech that does not engage the vocal folds or produce any acoustic output and is indiscernible to an external observer. In simple words, any form of speech in absence of the first two steps of speech production that is characterized by miniscule movements in larynx and articulatory muscles is termed as internal articulation. This mode of speaking naturally occurs while reading that helps the human brain to comprehend what is being read and potentially reduces the cognitive load on the brain.

It should be clearly established that the internal articulation process is completely voluntary and occurs in the articulatory system upon receiving the stimulus from the brain. It is often confused with the thought process involved in the production of speech that occurs in the brain itself. The brain is responsible for the selection of words, organization of relevant grammatical forms and control of the motor system associated with the vocal apparatus (shown in figure 3-6). The motor system actuates the vocal apparatus accordingly and produces speech whether it be acoustic, whispered or silent (internally articulated). Internal articulation occurs due to the cumulative action of the brain and the peripheral nervous system but is very different from the speech conception happening in the brain.

4. MATHEMATICAL MODELING

Development of precise hardware as well as software in any system embodies proper calculation of design parameters which can be done using ideal mathematical models.

4.1 Filter Design

This section includes the mathematical models for the parametric calculations of hardware components along with circuit implementation specifications.

4.1.1 Analog Butterworth Filter

Butterworth filter is a signal processing filter designed to have a frequency response as flat as possible in the passband i.e. ideally no ripples in the pass band. The transfer function of an ideal n-order Butterworth filter is given by

$$|T_n(\mathrm{jw})^2| = \frac{1}{\left(1 + \frac{w_0}{\mathrm{w}}\right)^{2\mathrm{n}}} \qquad 4.1$$

Where, w_o = cut-off frequency of filter

The frequency response of the Butterworth filter is obtained from above equation which is as shown in figure below

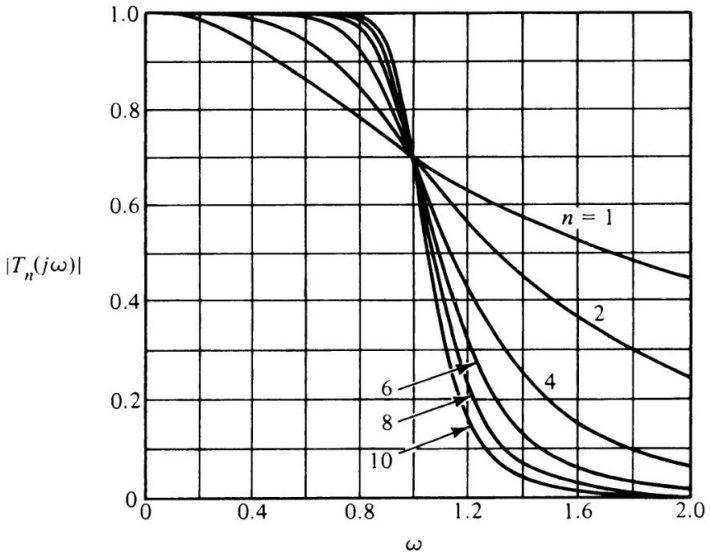

Figure 4-1: Frequency Response of Different Orders of Butterworth Filter

The attenuation (A) introduced by a Butterworth filter is given by

$$A = 20\log|T(jw)|dB \quad 4.2$$

For the pass band extending from w = 0 to w = w_p the attenuation should not exceed Amax. From w_p to w_s lies transition band and for a stop band beyond w_s the attenuation should not be less than A_{min}.

The order of a Butterworth filter with maximum attenuation for passband (A_{max}), minimum attenuation for stop band (A_{min}) is calculated using the following equation

$$T(jw) = \frac{T_0}{1 + \varepsilon^2 \left(\frac{w_s}{w_p}\right)^{2n}} \quad 4.3$$

Where ε = maximum pass band gain,
w_s = stop band frequency,
w_p = pass band frequency.
The order of the Butterworth filter also determines its roll-off characteristics. For a Butterworth filter of order 'n' the roll-off rate is 20n dB/decade or 6n dB/octave. The design and circuit implementation of such higher order Butterworth filters with all the above parameters can be done in different filter design topologies such as Cauer topology, Sallen and Key topology, etc. [15]

4.1.1.1 High Pass Filter

High-pass filter passes signals above cut-off frequency (f_c) and attenuates signals lower than the cut-off frequency. Based on the design requirements the order of a Butterworth high-pass filter can be determined from equation 4.1. General high-pass passive filter of the second order is as shown in the figure below.

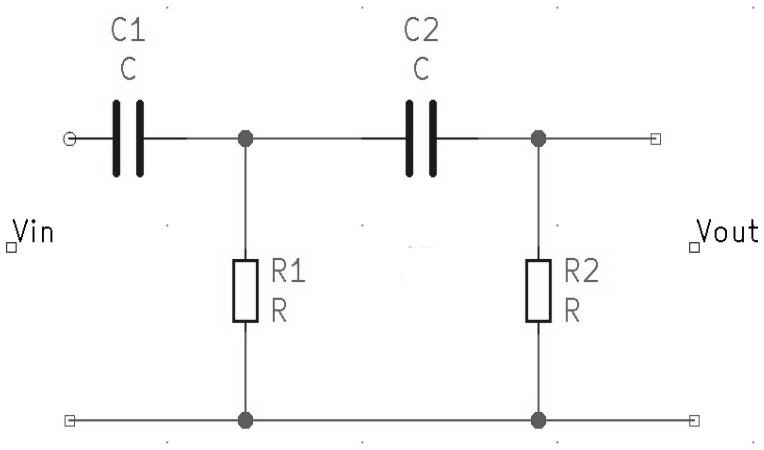

Figure 4-2: Second Order Passive Butterworth HPF

The cut-off frequency of the above filter circuit is given by

$$f_c = \frac{1}{2\pi\sqrt{R_1 R_2 C_1 C_2}} \tag{4.4}$$

Let $R_1 = R_2 = R$ and $C_1 = C_2 = C$ the above equation simplifies into

$$f_c = \frac{1}{2\pi RC} \tag{4.5}$$

The value of C is determined as per the choice or given cut-off frequency and the corresponding value of R is determined from the above equation.

4.1.1.2 Low Pass Filter

Low-pass filter passes signals below cut-off frequency (f_c) and attenuates signals higher than the cut-off frequency. Based on the design requirements the order of a Butterworth low-pass filter can be determined from equation 4.1. Sallen and Key topology of active analog filter design is very common in practice. Basic 2nd order LPF based on this topology as shown in the figure below.

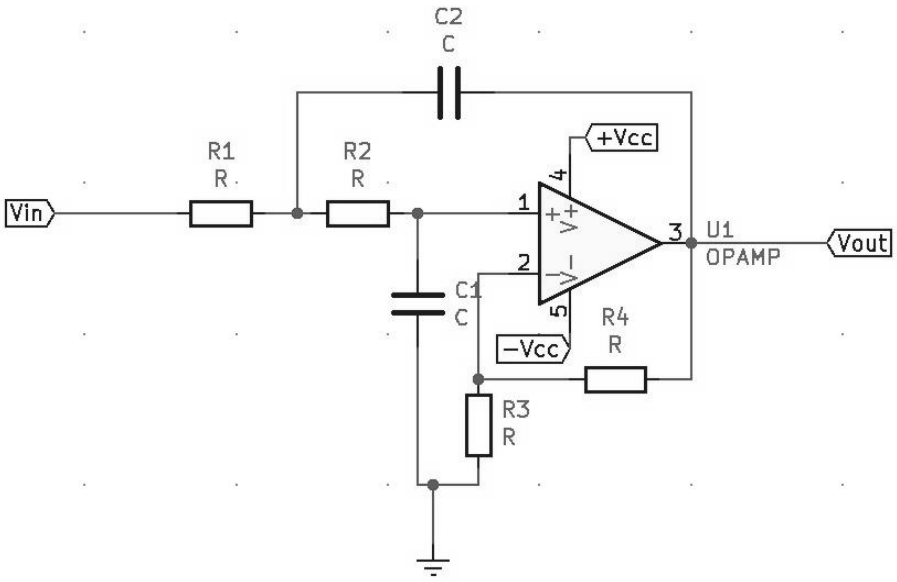

Figure 4-3: General Second Order Sallen-Key LPF

The transfer function of the above circuit is given by

$$\mathrm{A(s)} = \frac{A_0}{1 + w_c\{C_1(R_1 + \mathrm{R}_2) + (1 - A_0)R_1C_2\}\mathrm{s} + {w_c}^2R_1R_2C_1C_2 2\mathrm{s}^2} \qquad 4.6$$

Where w_c is the cut-off frequency

Since $R_1 = R_2 = R$ and $C_1 = C_2 = C$ we have

$$\mathrm{A(s)} = \frac{A_0}{1 + w_c\mathrm{RC}(3 - A_0)s + (w_cRC)^2} \qquad 4.7$$

$$A_0 = 1 + \frac{\mathrm{R}_4}{\mathrm{R}_3} \qquad 4.8$$

Consider a and b as,

$$\mathrm{a} = w_cRC(3 - A_0) \qquad 4.9$$

$$\mathrm{b} = (w_cRC)^2 \qquad 4.10$$

The value of C is set as per the choice and the corresponding values of R and A_0 are determined using the following equations.

$$R = \frac{\sqrt{b}}{2\pi f_c C} \quad 4.11$$

$$A_0 = 3 - \frac{a}{\sqrt{b}} \quad 4.12$$

$$A_0 = 3 - \frac{1}{Q} \quad 4.13$$

Where Q = pole quality of the filter

On comparing the denominator of transfer function with 2nd order Butterworth polynomial which is,

$$B_2(s) = s^2 + 1.414s + 1 \quad 4.14$$

Which gives,

$$1 = \frac{1}{w_c RC} \quad 4.15$$

$$w_c = \frac{1}{RC} \quad 4.16$$

$$f_c = \frac{1}{2\pi RC} \quad 4.17$$

Where f_c is the cutoff frequency of the filter, $w_c = 2\pi f_c$ and,

$$1.414 = \frac{3 - A_0}{w_c RC} \quad 4.18$$

From equation 4.17,

$$1.414 = 3 - A_0 \quad (4.19)$$

$$A_0 = 1.586 \quad (4.20)$$

From equations 4.13 and 4.20

$$Q = 0.707 \quad (4.21)$$

The value of Q must be equal or near to 0.707. Otherwise, if the value of Q is greater in a 2nd order filter, it will respond to a step input by quickly rising above, oscillating around, and eventually converging to a steady-state value. With a very low quality factor it will respond to a step input by slowly rising toward an asymptote. The response being similar to 1st order low pass filters [16].

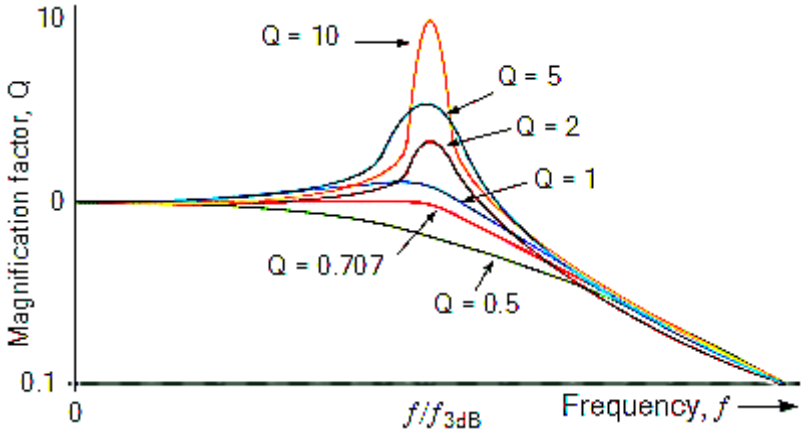

Figure 4-4: Q factor vs Frequency

4.1.2 Digital Filters

The transfer function of a digital filter is derived from that of analog filters. Different methods such as bilinear transformation, impulse invariance method, etc can be used for transformation. The transfer function of a typical digital filter is given by,

$$H(e^{jw}) = \frac{B(e^{jw})}{A(e^{jw})} \quad (4.22)$$

$$H(e^{jw}) = \frac{(b_0 + b_1 e^{-jw} + \cdots + b_M e^{-jwM})}{(a_0 + a_1 e^{-jw} + \cdots + a_N e^{-jwN})} \quad 4.23$$

Where M and N are the order of numerator and denominator respectively.

4.1.2.1 Butterworth Filter

The passband [f_{pa}, f_{pb}] of a digital bandpass filter can be mapped onto the entire passband [$-\Omega_{pass}$, Ω_{pass}] of the analog filter such that,

$$-\Omega_{pass} = \frac{c\ - \cos(\omega_{pa})}{\sin(\omega_{pa})} \quad 4.24$$

$$\Omega_{pass}\ = \frac{c\ - \cos(\omega_{pb})}{\sin(\omega_{pb})} \quad 4.25$$

where $\omega_{pa} = \frac{2\pi f_{pa}}{f_s}$,

$\omega_{pb} = \frac{2\pi f_{pb}}{f_s}$ and

$c\ = \frac{\sin(\omega_{pa} + \omega_{pb})}{\sin \omega_{pa}} + \sin \omega_{pb}$.

Note that for ω_{pa}, ω_{pb} in the interval [0, π], the above expression for c implies $|c| \leq 1$, as required for stability.

Similarly, the stopband [Ω_{sa}, Ω_{sb}] of the digital filter should map exactly onto the stopband [$-\Omega_{stop}$, Ω_{stop}] of the analog filter. As the Butterworth magnitude response is a monotonically decreasing function of Ω, it is enough to choose the smallest of the two stopbands i.e.,

$$\Omega_{stop} = \min(|\Omega_{sa}|, |\Omega_{sb}|) \quad 4.26$$

With corresponding values of Ω_{pass} and Ω_{stop} the Butterworth parameters N and Ωo are computed as,

$$H_i(z) = \frac{1}{\frac{1 - 2\cos\theta_i * s}{\Omega_0} + \frac{s^2}{\Omega_0}} \qquad 4.27$$

$$H_i(z) = \frac{1 - 2cz^{-1} + z^{-2}}{1 - z^{-2}}$$

$$\text{or } H_i(z) = \frac{G_i(1 - z^{-2})^2}{1 + a_{i1}z^{-1} + a_{i2}z^{-2} + a_{i3}z^{-4} + a_{i4}\,z^{-4}} \qquad 4.28$$

where $G_i = \frac{\Omega_0^2}{1 - 2\Omega_0 * \cos\theta_i + \Omega_0^2}$ for i = 1, 2,...,K.

4.1.2.2 Notch Filter

The transfer function of digital notch filter is computed by bilinear transformation of its respective analog filter similar to that of the digital bandpass filter.

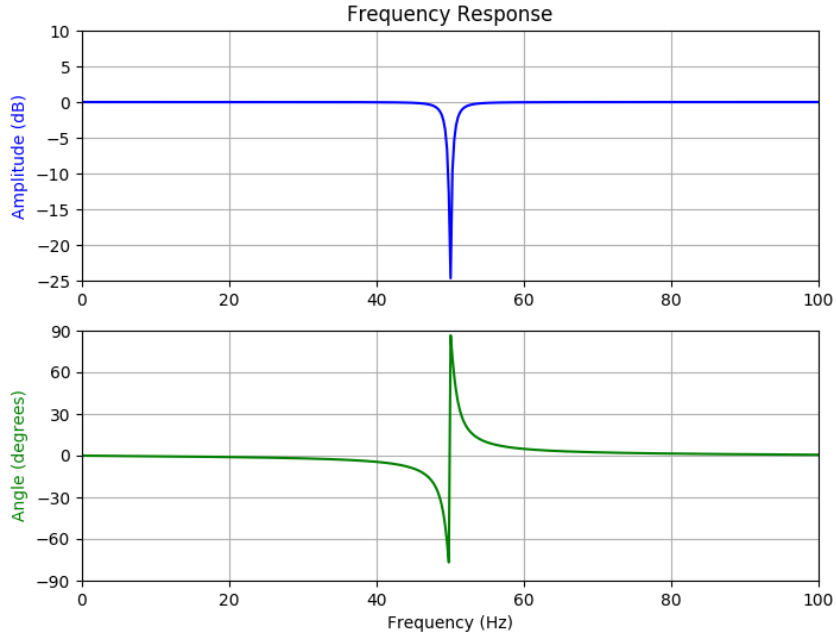

Figure 4-5 : Response of Notch Filter

4.2 Noise and Artifacts

Noise is any unwanted disturbance that hinders or interferes with a desired signal. To put it differently, everything that is not part of the signal wanted to be measured is considered noise. However, a differentiation can be made between disturbances or interferences and the word noise. Disturbances often come from sources external to the circuit under study, and result from electromagnetic or electrostatic coupling with the power lines, fluorescent

lights, cellphones and cross-talk between adjacent circuits, even mechanical vibration could cause disturbances. These types of disturbances and possible sources of interference are mostly “man-made” and can be minimized or eliminated.

4.2.1 ECG Artifacts

EMG signal extract is bound to be contaminated by the electrical activity from the heart. The placement of EMG electrodes, which is conducted by a selection of the pathological muscle group, often decides the level of ECG contamination in EMGs. Due to an overlap of frequency spectra of ECG and EMG signals and their relative characteristics, it is very difficult to remove the ECG artifacts from the EMG signal.

ECG contamination in EMGs may be kept at a minimal level by common-mode rejection at the recording site, through careful placement of bipolar recording electrodes along the heart's axis if possible. The effects ECG artifacts can be minimized using an appropriate wavelet which can be used to mimic the heart beat pulses. Some of them are Ricker wavelet, Sym8 wavelet, Morlet wavelet, etc. [17]

Ricker wavelet also known as Mexican Hat wavelet is a pulse-shape signal and is generally used in determining the pulse period and mimicking the impulsive portion of the pulse. It is the second derivative of the Gaussian wavelet also known as Laplacian of Gaussian function. Ricker wavelet in time domain is given by,

$$R(f) = (1 - 2\pi^2 f^2 t^2) e^{-\pi^2 f^2 t^2} \qquad 4.29$$

The frequency spectrum of this wavelet is real and non-negative, $|R(f)| = R(f)$. Also the differentiation of the function in frequency domain is zero i.e.: $dR(f)/df = 0$. Thus any delay in time domain only affects the magnitude of the function while the phase remains constant which makes the detection of the impulses convenient [18]. A typical Ricker wavelet with width and scaling parameters τ and σ respectively is as shown in figure below.

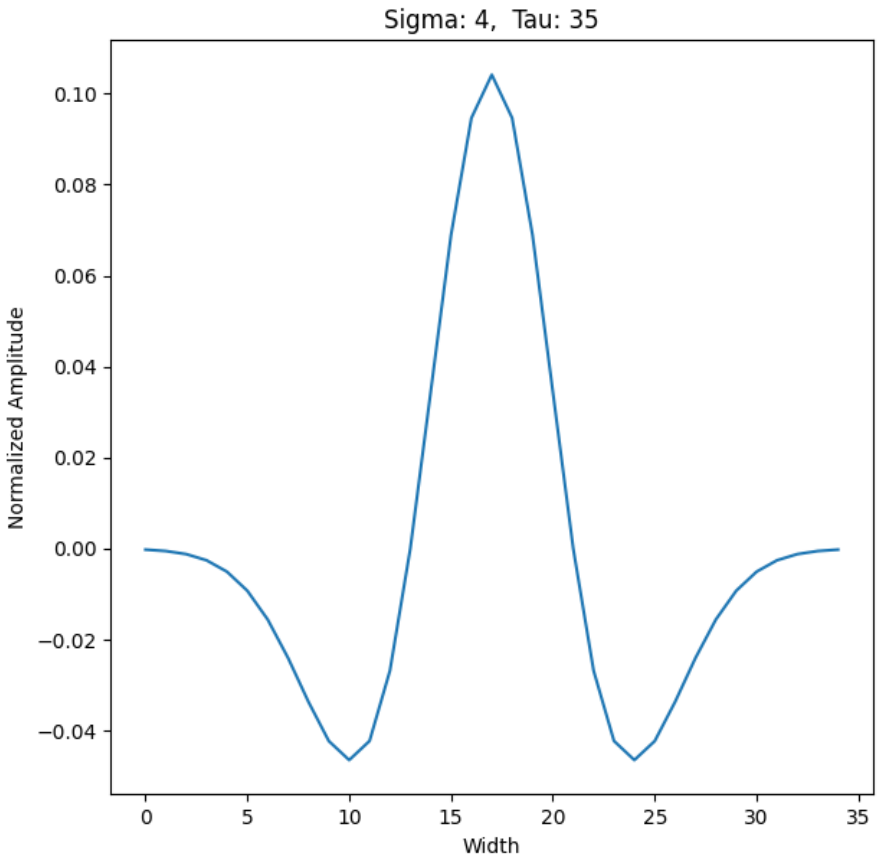

Figure 4-6 : Ricker Wavelet

The EMG signal is convoluted with the Ricker wavelet to get the ECG signals. The obtained ECG signal is then removed from the original signal without losing much of its features.

4.2.2 Electromagnetic Noise

The human body behaves like an antenna; the surface of the body is continuously inundated with electric and magnetic radiation, which is the source of electromagnetic noise. Electromagnetic sources from the environment superimpose the desired signal, or cancel the signal being recorded from a muscle. The amplitude of the ambient noise (electromagnetic radiation) is sometimes one to three times greater than the EMG signal of interest.

The dominant concern for the ambient noise arises from the 50 Hz radiation from power sources, which is also called line noise. This is caused by differences in the electrode impedances and in stray currents through the patient and the cables. However, in order to remove the recorded artifact, off-line processing is necessary. Line noise, n(t) with its harmonics can be mathematically represented as:

$$n(t) = \cos(2\pi * 50t) + \cos(2\pi * 100t) + \cos(2\pi * 200t) + \cos(2\pi * 300t) \quad 5.1$$

A number of adaptive filter techniques have been proposed for the attenuation of the line noise, such as adaptive FIR notch filter, adaptive IIR notch filter, adaptive notch filter using Fourier transform and so forth. These filters improve the SNR of an EMG signal by eliminating the line noise from the system.

4.3 Feature Extraction

Most of the suitable modern neural networks are able to learn the features from the provided EMG signals, but the black-box nature of deep learning does not reveal much information learned by the network and how it relates to handcrafted features. The high variability of EMG recording between participants causes neural networks to generalize poorly across subjects using standard training methods. Therefore, a hybrid approach of providing handcrafted features for a deep learning model stands to be more suitable for training a neural network in this regard [19]. These handcrafted features include both the temporal and spectral features.

4.3.1 Temporal Features

Temporal features depicts the characteristics of the signal in time domain which typically includes amplitude, onset and power. These attributes can be studied to represent the signal properties in a comprehensive manner.

4.3.1.1 Zero Crossing Rate

Zero Crossing Rate (ZCR) is the rate of sign changes of a signal i.e. the rate of which the signal changes from positive to zero to negative or from negative to zero to positive.

This is a temporal feature used in both speech recognition and music information retrieval. ZCR is defined formally as:

$$ZCR = \frac{1}{T-1}\sum_{t=1}^{T-1} 1_{R<0}\,(S_t S_{t-1}) \qquad 4.31$$

Where **S** is the signal of length **T** and $\mathbf{1_{R<0}}$ is an indicator function.

4.3.1.2 Nine Point Double Average

The nine-point double-averaged signal w[n] is defined as

$$w[n] = \frac{1}{9}\sum_{k=-4}^{4} x[n+k] \qquad 4.32$$

Where v[n] is given by,

$$v[n] = \frac{1}{9}\sum_{k=-4}^{4} x[n+k] \qquad 4.33$$

Where x[n] is the mean of an EMG signal.

4.3.1.3 High Frequency Signal

The high frequency signal is obtained by subtracting the nine point double average from normalized mean. Formula for calculating high frequency and rectified high frequency is given by 4.34 and 4.35 respectively.

$$p[n] = x[n] - w[n] \qquad 4.34$$

$$r[n] = |p[n]| \qquad 4.35$$

4.3.1.4 Frame Based Power

The Frame based power is the sum of squares of the signal in the frame. Which is given as,

$$P_f = \sum_{i=0}^{k-1} (a[i])^2 \quad 4.36$$

Where k is the frame size and a[i] is a segment of the signal.

4.3.2 Spectral Features

Spectral features imply the frequency domain (spectrum) parameters of data. When dealing with analysis of signals, frequency domain representation becomes more appropriate as it allows for observation of signal characteristics that is not easily seen in time domain.

4.3.2.1 Short Time Fourier Transform

Short Time Fourier Transform (STFT) is obtained by introducing a sliding window to the time variant signal so it is also known as time-dependent Fourier transform. This window adds a new dimension of time to the frequency response by suppressing the input signal outside a certain region.

There are different types of windowing functions that can be applied depending on the characteristics of a signal. Hamming, Hanning, Blackman-Harris and Gaussian are some examples. The frequency domain plot of a window is a continuous spectrum with a main lobe centered at each frequency component of time domain signal and side lobes approaching zero. Lower amplitude of the side lobes reduces the spectral leakage, a phenomenon in which the spectrum is smeared due to leakage of energy from the frequency components nearby. It can further be reduced by increasing the roll-off rates of side lobes.

Let us consider X(t) be a time domain signal which is to be fed to the window characterized by a transfer function W(t), then the output of the window is given by,

$$Y(t) = X(t)W(t) \qquad 4.37$$

Here W(t) truncates the signal beyond its window size resulting in only a part of the input signal as Y(t). The cut-off points of W(t) introduce high frequency components at the beginning and at the end of the output Y(t). The input signal applied with the Hamming window is as shown in the figure below.

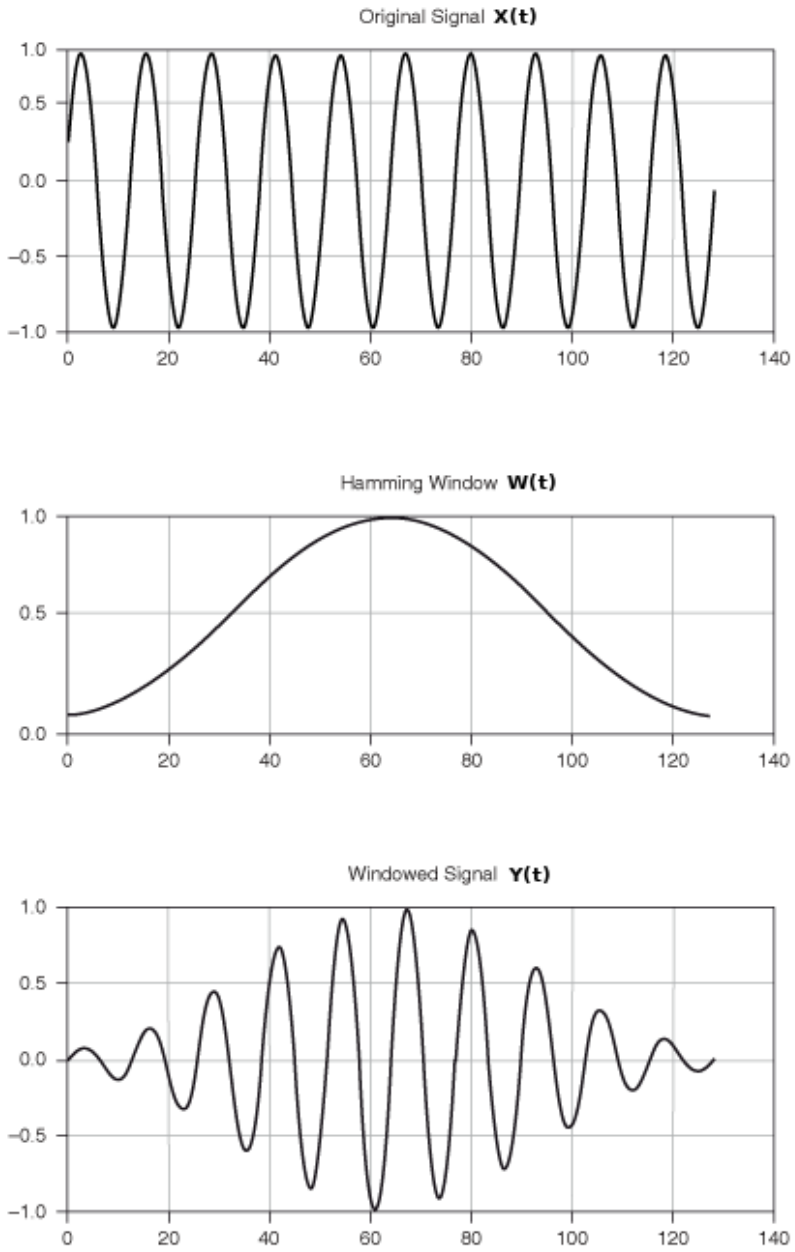

Figure 4-7 : Hamming Window Applied to an Input Signal

Let us consider an input signal x(t) is introduced to the sliding window with window function $\gamma(t)$ then the discrete time STFT of the signal is given by,

$$X(\tau, \omega) = \int_{-\infty}^{\infty} x(t)\gamma(t-\tau)e^{-j\omega n}\, dt \qquad 4.38$$

Here $\gamma(\tau)$ is the window interval centered at zero.

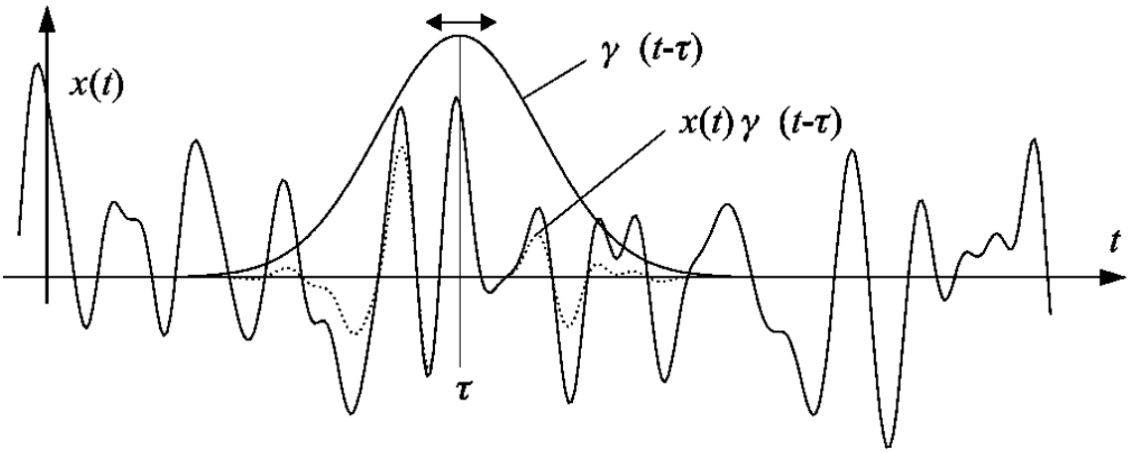

Figure 4-8 : STFT of a Continuous Time Signal

Let us consider, Δt be the radius of the time window γ(t) centered at τ(0) while Δω be the frequency window **Γ** (ω) centered at w_0. Form Heisenberg uncertainty principle we obtain,

$$\Delta t * \Delta\omega \geq \frac{1}{2} \qquad 4.39$$

This relation shows that size of time-frequency windows cannot be made arbitrarily small and that a perfect time-frequency resolution cannot be achieved [20] . Thus, the selection of window size should be done considering equation 4.39. The window function is assumed to be non-zero only within the window interval. The time-frequency resolution of the spectrogram will be dependent on the chosen value of window size. Large window size results in very low time-frequency resolution while very small window size cannot locate the time domain so the selection of appropriate window size is essential. The selection of appropriate window size depends on the type of signal [21]. Also, the type of window selection depends on the type of signal.

The squared magnitude of STFT is known as a spectrogram. Generally, STFT is complex-valued so spectrograms are often used for further processing of the signal. The equation for spectrogram S(τ,ω) can be obtained by rectifying and then squaring equation 4.40 as,

$$S(\tau, \omega) = |\int_{-\infty}^{\infty} x(t)\gamma(t-\tau)e^{-j\omega n}\, dt|^2 \qquad 4.40$$

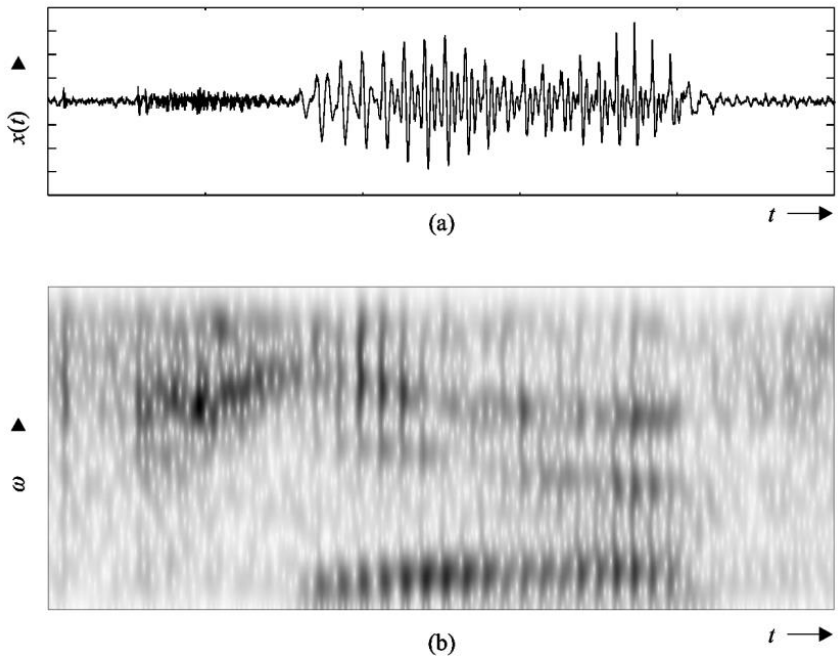

Figure 4-9: a) Input Speech Signal b) Spectrogram of the Input Signal

5. INSTRUMENTATION AND REQUIREMENT ANALYSIS

The hardware as well as software requirements are to be analyzed before implementation. The analysis includes specifications, responses and supportive environment of the component in the system.

5.1 Hardware Components

This section includes hardware component description along with the analysis of individual response in the circuit.

5.1.1 Electrodes

An electrode is a solid electric conductor through which an electric current enters or leaves an electrolytic cell. It is simply a transducer that converts ionic potentials to electric potentials. There exist two main types of electrodes for the extraction of EMG signals from the body. They are:

5.1.1.1 Indwelling Electrode

Indwelling electrodes are invasive electrodes inserted through the skin directly over the muscle. Needle electrodes and fine wire electrodes are two commonly used indwelling electrodes used to measure action potential of a motor unit directly. Indwelling electrodes have two main advantages. One is that its relatively small pickup area enables the electrode to detect individual MUAPs during relatively low force contractions. The other is that the electrodes may be conveniently repositioned within the muscle (after insertion) so that new tissue territories may be explored. However, better selectivity and crosstalk immunity of indwelling electrodes comes at a price. They are painful and carry the risk of infections. [22]

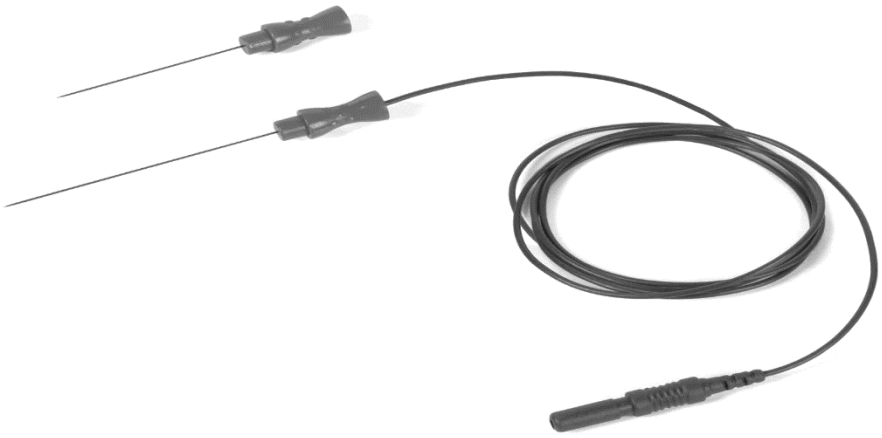

Figure 5-1: Monopolar Needle Electrode

5.1.1.2 Surface Electrode

Surface electrodes are non-invasive electrodes placed on the skin directly over the muscle for measurement and detection of EMG signal. These electrodes are simple and very easy to implement and do not require medical supervision and certification. It is designed to selectively obtain the surface EMG signal while minimizing the artifacts, DC potentials and environment noise picking.

The theory behind the working of surface electrodes is that they form a chemical equilibrium between the detecting surface and the skin of the body through electrolytic conduction, so that current can pass from an electrolyte to a non-polarized electrode oxidizing the electrode atoms. The resulting cations and electrons flow in opposite directions: the electrons go through the metal cables attached to the electrodes meanwhile the cations go to the electrolyte. However, use of proper electrolytes with respective electrodes should be ensured for the electrolytic conduction to occur. [22]

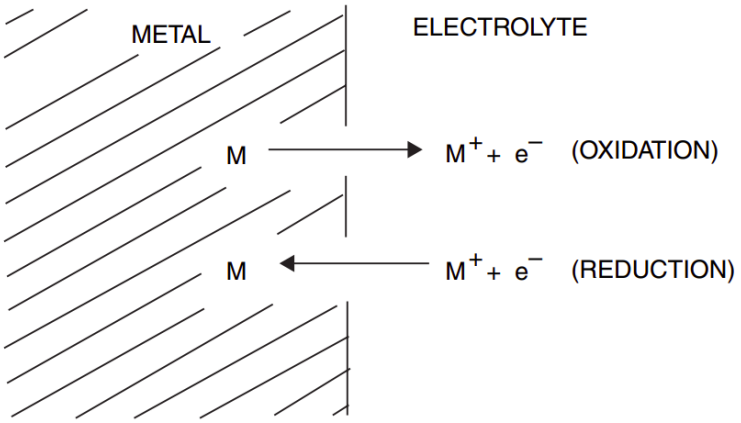

Figure 5-2: Electrode-electrolyte Interface

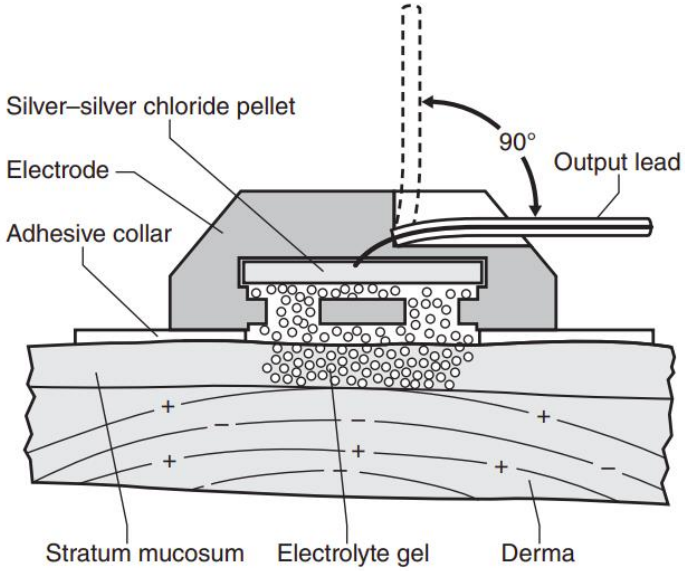

Figure 5-3: Skin-Electrode Interface (Ag-AgCl electrodes)

When the electrochemical reaction between the metal and the electrolyte stabilizes, a potential difference known as "half-cell potential" is formed between the negative electrode and the positive electrolyte which is determined by the Nernst equation as:

$$E = \frac{RT}{nF} ln(\frac{a_1}{a_2}) \quad 5.1$$

Where a_1 and a_2 are ionic activities on each side of the membrane,

E= half-cell potential,

R= universal gas constant = 8.314 Joule per mole per kelvin,

T= absolute temperature,

n= the number of valence electrons in the metal,

F= Faraday Constant = 96485 C per mole.

The half-cell potential of a single electrode results in a DC offset in EMG signal. If two chemically identical electrodes make contact with the same electrolyte/body, the two interfaces should, in theory, develop identical half-cell potentials. When connected to a differential amplifier, the half-cell potentials of such electrodes would cancel each other out and the offset voltage would be zero. The electrode potentials would, therefore, make zero contribution to a bio-signal they were being used to detect. Unfortunately, slight differences in electrode metal or gel result in the creation of offset voltages, which can greatly exceed the physiological variable to be measured. Generally, a more significant problem is that the electrode offset voltage can fluctuate with time, thus distorting the monitored bio-signal.

The skin, gel, and electrode interfaces function as a complex physical system that is frequency dependent and affects the EMG signal in a deterministic way. It represents a complex impedance that can be modeled as a capacitor (C_1) in series with a resistor (R_1). This impedance may vary from a few kilo-ohms to a few mega-ohms, depending on electrode size and skin condition. There is an additional resistor (R_2) in parallel to denote the resistance of the chemical reaction (activation energies) that moves the charge at the interface to accurately model the skin-electrode interface.

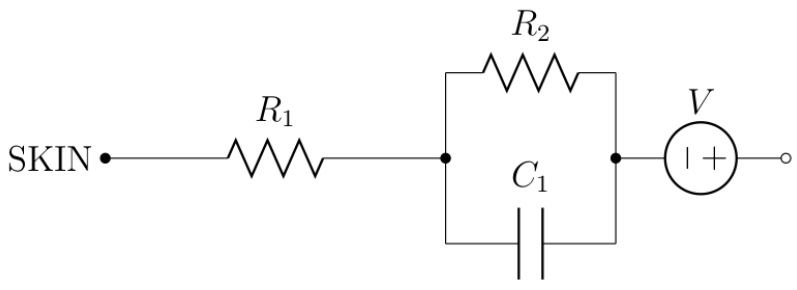

Figure 5-4: Skin-Electrode Circuit Model

Where V = Half-cell potential

C_1 = capacitive effects of the electrolyte dipole layer at the electrode surface,
R_1 = Bulk resistance of the electrolyte gel,
R_2 = resistance of the chemical reaction (activation energies) that moves the charge at the interface.

Surface electrodes are usually made up of silver/silver chloride (Ag-AgCl), silver chloride (AgCl), silver (Ag) or gold (Au) or platinum (Pt). Surface area, utility, selectivity, sensitivity and many other parameters vary with the type of material used in the electrode. Selecting the proper type of electrodes that can result in having low electrode-skin impedance and can last longer for recording is important for EMG measurements.

Surface electrodes can be either polarizable or non-polarizable. The electrode where no actual charge crosses the electrode-electrolyte interface when a current is applied is a polarizable electrode. The current across the interface is a displacement current and the electrode acts like a capacitor. The electrode where the current passes freely across the electrode-electrolyte interface without any external energy to make the transition is a non-polarizable electrode. Platinum electrode is an example of polarizable electrode whereas Ag-AgCl electrode is an example of non-polarizable electrode. [22]

A. Silver−Silver Chloride Electrodes

Ag-AgCl electrodes are electrodes with a thin layer of silver coating on plastic substrates and the outer layer of silver is converted to silver chloride. Some of the important characteristic of Ag-AgCl electrodes are:

- Low half-cell potential of about 220 mV
- High conductivity of 6.30×10^7 Siemens per meter at 20℃
- High exchange current density of 10A/cm
- Low level of intrinsic noise
- Low contact impedance

Electrodes made of Ag-AgCl are often preferred over the others, as they are almost non-polarizable electrodes, which means that the electrode-skin impedance is resistive and

not capacitive. Low half-cell potential results in low DC offset in recordings and small redox potential facilitates the easier and fast exchange of ions. Therefore, the surface potential is less sensitive to relative movements between the electrode surface and the skin. Additionally, these electrodes provide a highly stable interface with the skin when electrolyte solution is interposed between the skin and the electrode. Such a stable electrode-skin interface ensures high signal to noise ratios, reduces the power line interference in bipolar derivations (50 Hz or 60 Hz frequencies and their harmonics) and attenuates the artifacts due to body movements. [22]

Figure 5-5: Disposable Ag-AgCl Electrodes

B. Gold Electrodes

Gold electrodes are electrodes with a thin layer of gold coating on metals like silver or copper. Some of the important characteristic of gold electrodes are:

- Half-cell potential of about 1.680 V
- Has high conductivity of 4.1×10^7 Siemens per meter at 20℃
- Higher contact impedance than Ag-AgCl
- Although expensive, they are reusable and durable
- High immunity to external noises

Typically, gold plated EMG electrodes have a 1.45 mm diameter conductive area on a disc of 10 mm. Smaller area provides high selectivity and thus is suitable for detection of EMG signals of a localized area or an individual muscle tissue.

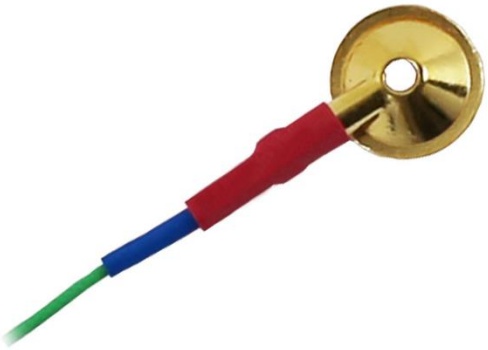

Figure 5-6: Gold Plated Cup Electrode

5.1.2 Electrolyte

Electrolyte in surface electromyography is a conductive gel that reduces the impedance of skin-electrode interface and facilitates the conduction of EMG signals. Different kinds of electrodes require different kind electrolytes for their use. The gold cup electrodes used in this project are used with Ten20 conductive paste manufactured by Weaver and Company.

The target skin area is first cleansed using isopropyl alcohol to remove skin oils, dead skin, sweat and dirt and then the electrode with ample Ten20 conductive paste is attached to the skin to extract the EMG signals. The amount of Ten20 paste be such that it should not interfere with the potential of the induced EMG signals.

Ten20 Conductive Paste is a mixture of the following components: Polyoxyethylene 20 Cetyl Ether, Water, Glycerin, Calcium Carbonate, 1,2 Propanediol, Potassium Chloride, Gelwhite, Sodium Chloride, Polyoxyethylene 20 Sorbitol, Methylparaben and Propylparaben [23].

Figure 5-7: Conductive Paste

5.1.3 Electrode Configuration

Electrode configuration refers to the number of recording surfaces and their arrangement relative to muscle, tendon and bony surface. The two most common methods are:

5.1.3.1 Monopolar Configuration

Monopolar uses three electrodes E1, E2 and Ground. E1 is placed over the muscle itself where the EMG signal is to be extracted and also referred to as “active recording surface electrode”. E2 is placed on an electrically neutral location such as tendon and also referred to as “reference electrode” and Ground is placed on a bony surface distant to E1 and E2. This configuration is called monopolar because only one electrode (E1) is used to record the muscle activity.

For monopolar configuration, select muscle on the skin surface where the lowest possible electrical stimulation will produce a minimal muscle twitch. The main drawback of this configuration is that it does not take full advantage of the differential amplifier design to reduce the unwanted noise in the EMG recordings.

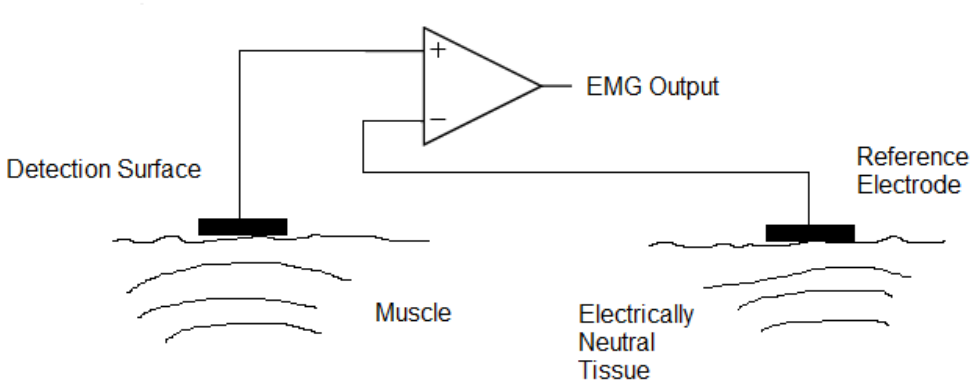

Figure 5-8: EMG Signal Extraction in Monopolar Configuration

5.1.3.2 Bipolar Configuration

Bipolar also uses three electrodes E1, E2 and Ground. E1 and E2 are placed over the muscle at a certain distance of about 5 to 20 mm apart. Ground is placed on a bony prominence typically near E1 and E2.

For bipolar configuration, a large enough muscle on the skin surface with lowest possible movement should be selected. This method overcomes the shortcoming of monopolar by taking the full advantage of amplifier circuitry that is designed to minimize unwanted interference signals from electromagnetic fields in the surrounding environment. However, the amplitude and frequency are largely dependent on the inter-electrode distance which sometimes is not easy to work with.

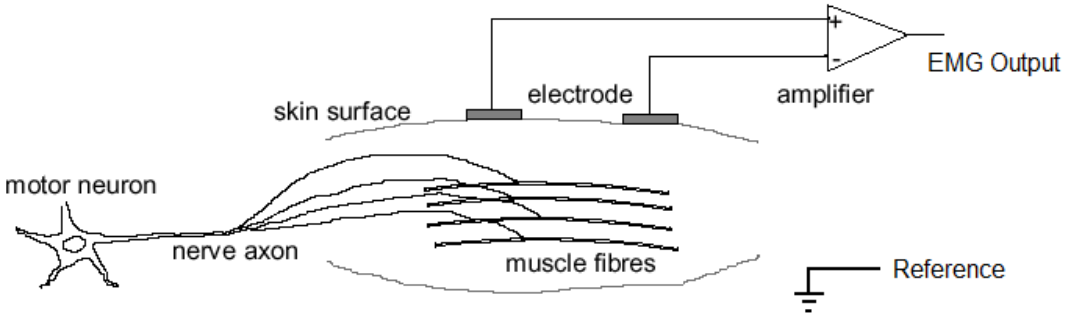

Figure 5-9: EMG Signal Extraction in Monopolar Configuration

5.1.4 Electrode Leads

Electrode leads are any set of wires that has the sole responsibility to transfer the charges induced on the electrodes to a signal acquisition system which has an amplifier (normally an instrumentation amplifier) at the front end. Simply, electrode leads are specialized cables designed to conduct electrical signals with minimum losses and distortion. Since signal from the electrode is fed to the amplifier through electrode leads, they are also termed as input leads. As the input leads offer a finite resistance, there will be some degree of voltage drop between the electrodes and amplifier resulting in loss of signal.

From ohm's law,

$$V_{drop} = I_{cable} \times R_{cable} \tag{5.2}$$

Resistance is a function of the conductivity of the material (σ), length (l) and surface area (A) which is given by,

$$R = lA \tag{5.3}$$

Of these three factors, length is the most critical because it can change to the greatest degree and is under control. Keeping the length of the input leads and all cables as short as possible will minimize the voltage drop.

A signal amplitude (V_{in}) is attenuated to differential voltage at the amplifier (V_{out}) by the electrode leads. Thus, the Attenuation (A) can be calculated as:

$$A = -20 \log\left(\frac{V_{out}}{V_{in}}\right) \tag{5.4}$$

The most sensitive part in the EMG system design is the path between the electrodes and the amplifier because it is where the EMG signal has the lowest voltage level and is most vulnerable to noise and interference pickup. The longer the signal has to travel, the more interference and noise get coupled electromagnetically. EMG signal for intended purpose

varies from 1 Hz to 500 Hz in frequency and is not susceptible to attenuation loss due dielectrics at high frequencies (above 1MHz). However, electromagnetic interference does occur and hinders the quality of propagating signals. This can be avoided by the use of shielded cables. Shielded cables are composed of three layers. A signal-carrying conductor at the center is covered by a flexible insulating layer, which is then surrounded by a braided metal sheath. Shielded cable acts as a Faraday cage to reduce electrical noise from affecting the signals. It also minimizes capacitive coupled noise from other electrical sources.

5.1.5 Cyton Board

The Cyton board is a hardware made by an open source community, OpenBCI, for extracting neuromuscular signals such as EMG, ECG and EEG. OpenBCI aims to improve research on neuromuscular signals, thus making the hardware portable and easier to use for enthusiasts and researchers.

OpenBCI Cyton board is powered with 3-6 V DC battery. It has 8 channel differential input for signal extraction. An onboard microcontroller is the brain of the device and performs the configuration tasks, controlling tasks and facilitates the communication between the multiple components of the board. It also comes along with a USB dongle which is used to interface the Cyton board with a remote computer. Both the Cyton board and USB dongle have low energy bluetooth for transmitting data wirelessly. [24]

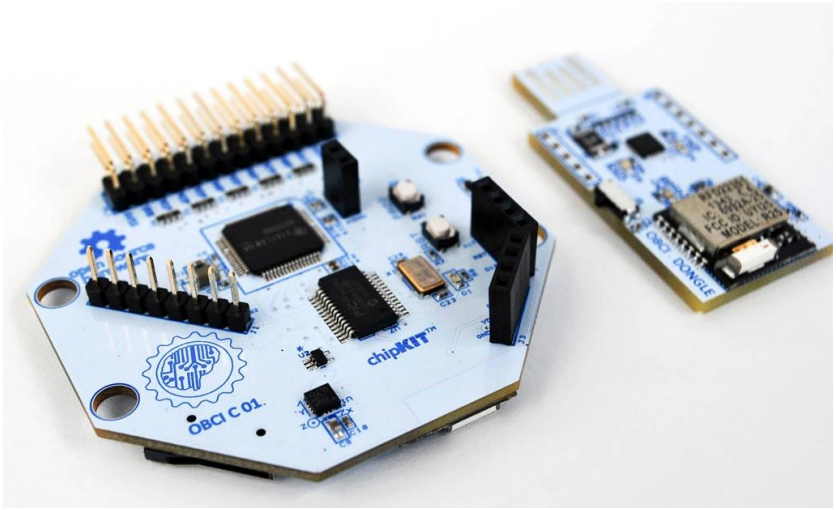

Figure 5-10: OpenBCI Cyton Board (Left) with USB Dongle (Right)

The Cyton board employs ADS1299, PIC32MX250F128B and RFduino to acquire and transmit data to a computer which has been further explained below:

5.1.5.1 ADS1299

It has ADS1299 Analog Front End, which has 8 channels, 24 bit ADC with a built-in programmable gain amplifier (PGA). The PGA gain can be set from one of seven settings (1, 2, 4, 6, 8, 12 and 24). ADS1299 incorporates all commonly-required features for EEG and ECG applications. With its high levels of integration and exceptional performance, ADS1299 enables creation of scalable medical instrumentation systems at significantly reduced size and power. Operating on voltage ranging from 4.75 - 5.25 volts. It has a flexible input multiplexer per channel that can be independently connected to an internally generated signal or test signal for temperature correction and lead off detection. The lead off detection is implemented internal to the device using an excitation current sink. It provides an SPI-compatible interface to communicate. Thus settings and parameters of ADS1299 can be changed by controller through the SPI interface.

5.1.5.2 PIC32MX250F128B

PIC32MX250F128B microcontroller is a 32 bit microcontroller that takes data from ADS1299 and sends it to a computer. Operating voltage of PIC32 ranges from 2.3V to 3.6V and has 128 KB of flash memory. It has I2S/SPI modules for codec and serial communication. It interfaces with a bluetooth module to send the data collected from ADS1299. It consists of firmware that sets the parameters of ADS1299 and controls the whole operation of the hardware.

5.1.5.3 RFD22301

RFD (RFduino) is a BLE (Bluetooth 4.0 Low Energy) module used for bluetooth communication between the Cyton board and USB dongle connected to the computer. The RFduino BLE module RFD22301 has a built-in ARM Cortex M0 microcontroller.

The operating voltage of this module lies between 2.1 V to 3.6 V. On-air data transmission rate varies between 250 kbps and 2000 kbps with receiver sensitivity of -39 dBm. Operating frequency of this module ranges from 2402 MHz to 2481 MHz with 1 MHz channel spacing.

5.2 Software Platforms

This portion includes the requirement analysis of development environments, libraries and circuit designing platforms.

5.2.1 OpenBCI GUI V5.0.1

OpenBCI GUI is a software tool for visualizing and recording EMG signals obtained from muscles. Signals can be displayed in real time, saved in the computer in .txt format. Real time and recorded data can be visualized as time series, Fast Fourier Transform (FFT), spectrogram plot and band power plot for individual channels. Furthermore, signals can also be live streamed to third party software like MATLAB. This standalone application is available on Windows, MAC and Linux. Data is saved into a different session folder every time the user initiates a new session.

5.2.2 Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together.

Python is also vastly used for data processing. Python has provided libraries like SciPy, pyEEG, pySpACE, bioSPPy for pre-processing of signals. Further processing of signals is done by using libraries like sklearn and librosa. Python also incorporates highly optimized open-source libraries like tensorflow, keras, numpy to affiliate model development and implementation. Python also provides GUI features using PyQt, it is a cross-platform GUI toolkit Qt, implemented as a Python plug-in. It provides an easy and effective platform for building GUIs.

5.2.3 KiCad

KiCad is a free software suite for Electronic Design Automation (EDA). It facilitates design of schematics for electronic circuits, converting them to printed circuit board

(PCB) designs along with circuit simulation. It has built-in footprints of different electronic components. Footprints of new components can also be designed manually.

5.3 Dataset

This portion includes comprehensive details regarding the dataset used throughout the project. During the early phase of the project, the speech EMG-UKA Trial Dataset prepared by Tanja Schultz and et al of Karlsruhe Institute of Technology, Germany was used. After successfully employing this dataset for word prediction models with considerable accuracy, a self-recorded dataset was prepared using Cyton Board.

5.3.1 Speech EMG-UKA Dataset

The EMG-UKA is a corpus of synchronous EMG and acoustic recordings of continuous speech collected for the purpose of subvocal speech recognition [25]. The dataset contains citations of at least 50 different sentences from an English News Broadcast per session in three modes of speaking: Audible, Whispered and Silent.

Audible speech is normally spoken speech with normal voicing and intonation. It is recorded using standard close-talking USB microphone sampled at 16 KHz. Whispered data is a speech emphasizing breath rather than vibration of the vocal tract. It is recorded using both the microphone similarly to the audible signal and EMG electrodes similarly to the silent speech. Silent speech is the speech with no sound while performing normal articulatory movements. It is recorded using the 7 channel EMG electrodes at a sampling frequency of 600 Hz. Channel 7 is just a marker signal that is used to synchronize different speaking modes in the data. The articulation muscles for extraction of these EMG signals are Levator Anguli Oris, Zygomatics Major, Platysma, Depressor Anguli Oris, Anterior Belly of Digastric and Tongue. It should be noted that the data is collected from a group of 4 speakers in sessions that are either multi-modal or single-modal. Here, multi-model specifies that the data is collected in two or three different modes at the same time using both the microphone and the EMG electrodes.

Table 5-1: Description of EMG-UKA Corpus

Mode	Sampling Rate (Hz)	Channels	Length (hh:mm:ss)	Speaker Count	Session Count
Audible	16000	2	01:52:24	4	6
	600	7	01:08:16	4	6
Whispered	600	7	00:21:47	4	6
Silent	600	7	00:22:21	4	6

The dataset arranges the audio data in ".wav" format and respective EMG data in ".adc" format. The corresponding sentence of the audio and EMG is transcribed in a sub-folder named as "Transcripts". Since there is some offset between audio and EMG signal, the dataset also provides the offsets and alignments of the uttered words. Moreover, the transcribed words are further broken down into their phonemes. The speakers for the dataset collection were all non-native speakers of English but were instructed well for clear pronunciation of each word and were between the age of 24 and 30 years old. Out of 4 speakers, 3 were male and 1 was female. [25]

5.3.2 Self-Recorded Dataset

A dataset was made using the Cyton board using all the available 8 channels in monopolar configuration. 8 signal electrodes were placed at different facial muscles involved during internal articulation as described in section 3.2.3.1. The ground electrode and reference electrode were attached to the ear lobes. Before placing the electrodes on the target muscles, the area was cleaned with isopropyl alcohol and a generous amount of electrolyte was applied to the electrodes that helped for both conduction and adhesion. The electrodes were further secured to the target muscles using medical tape.

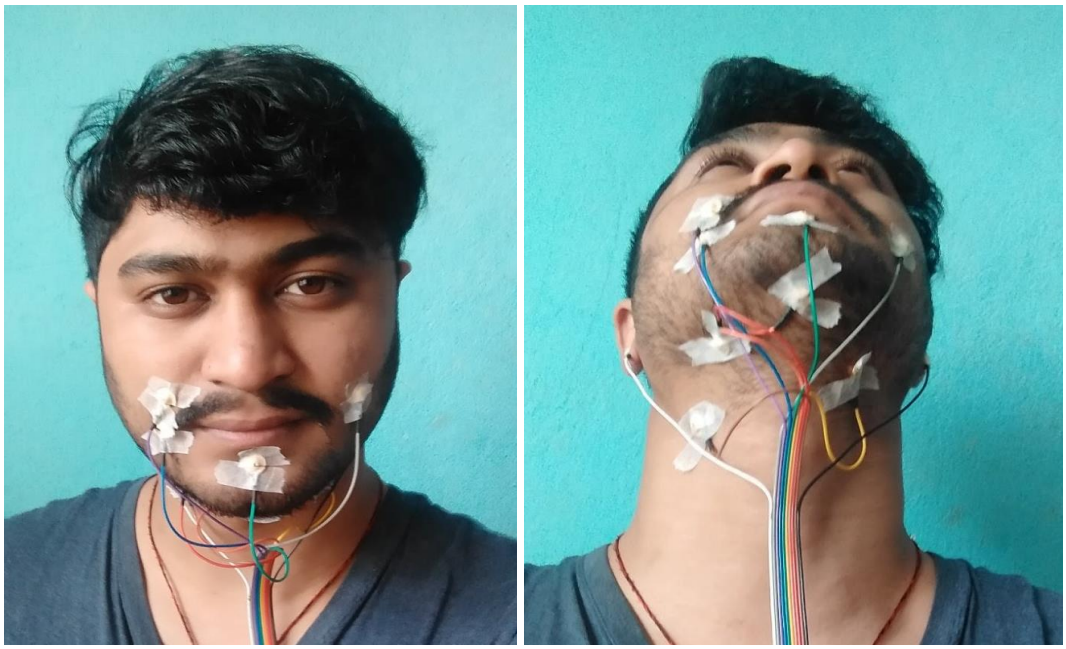

Figure 5-11: OpenBCI Electrode Placements (Front View)

Figure 5-12: OpenBCI Electrode Placements (Side View)

The selection of muscles was done according to the convenience of the electrode placement and the strength of the signals during internal articulation. The EMG signals from the muscles were extracted using gold cup electrodes (1.45 mm diameter conductive area) at a sampling rate of 250 Hz. A push button on the Cyton board was pressed for every sample collection. A total of 10 phonetically different frequently used words were

selected for the dataset. The extracted signals from all the channels were saved in a ".txt" file using OpenBCI GUI.

The dataset was recorded from 4 male subjects of ages ranging from 22 to 24 years old with an average age being 23 years old. The subjects were directed to speak in two different modes: 'Muscle Movement' and 'Mentally Rehearsed'. 'Muscle Movement' mode demanded the users to utter a word without making any sound but permissible mouth movements and in 'Mentally Rehearsed' mode, no mouth movement nor any sound was permissible. The recording environment was kept the same for all the speakers.

Table 5-2: Description of Self-Recorded Dataset

Mode	**Sampling Rate (Hz)**	**Channels**	**Length (hh:mm:ss)**	**Speaker Count**	**Session Count**
Muscle Movement	250	8	01:00:01	4	6
Mentally Rehearsed	250	8	01:31:59	4	6

The above table provides a summarized view of the dataset. The total length of the data is 2 hours and 32 minutes long. Out of 6 sessions, 3 recording sessions belong to one speaker and the other 3 sessions belong to respective speakers.

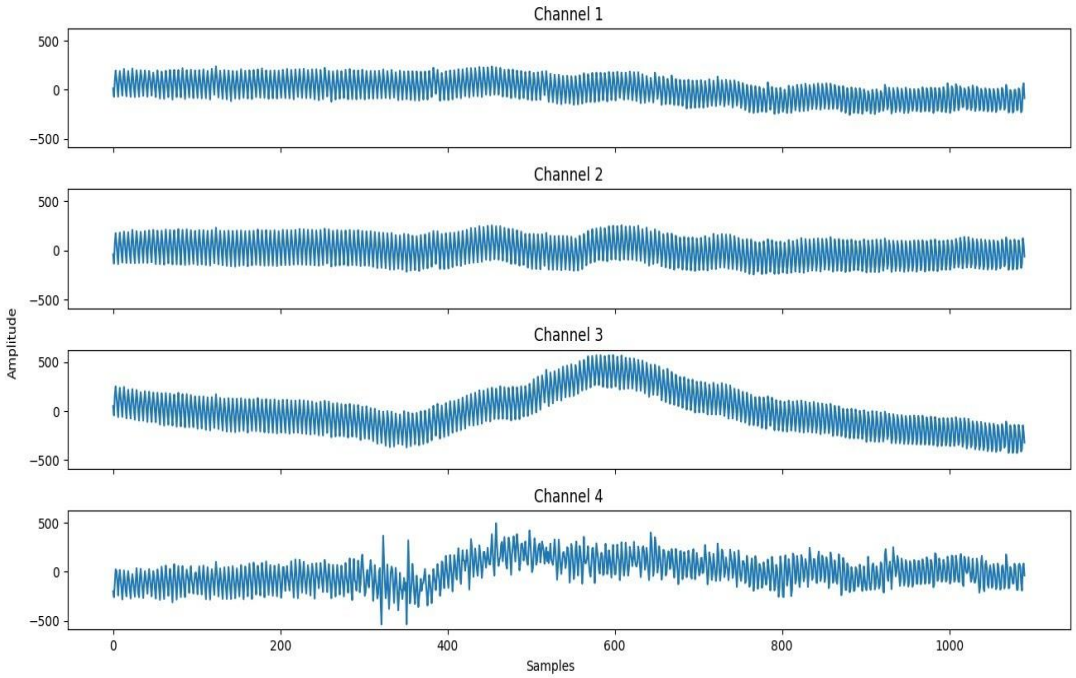

Figure 5-13: Raw Signal of Word “CALL” (Channel 1-4)

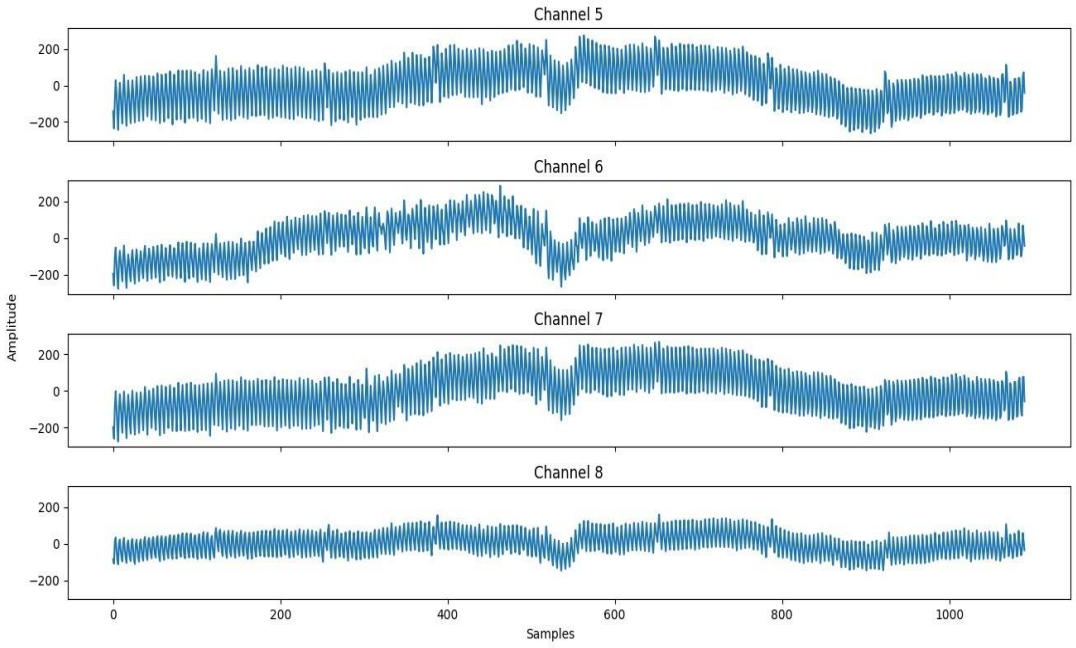

Figure 5-14: Raw Signal of Word “CALL” (Channel 4-8)

The feature within the OpenBCI enables to extract the instance of the utterance. The raw signals captured from the selected muscles for the utterance of the word “call” is as shown in above figure. Like this many utterances of different words are recorded for creating the dataset.

6. SYSTEM ARCHITECTURE AND METHODOLOGY

The architecture of the designed system has been represented graphically using figures and functional blocks. Signal extraction, amplification and processing mechanisms are illustrated in details along with the extraction of features and machine learning algorithms.

6.1 System Block Diagram

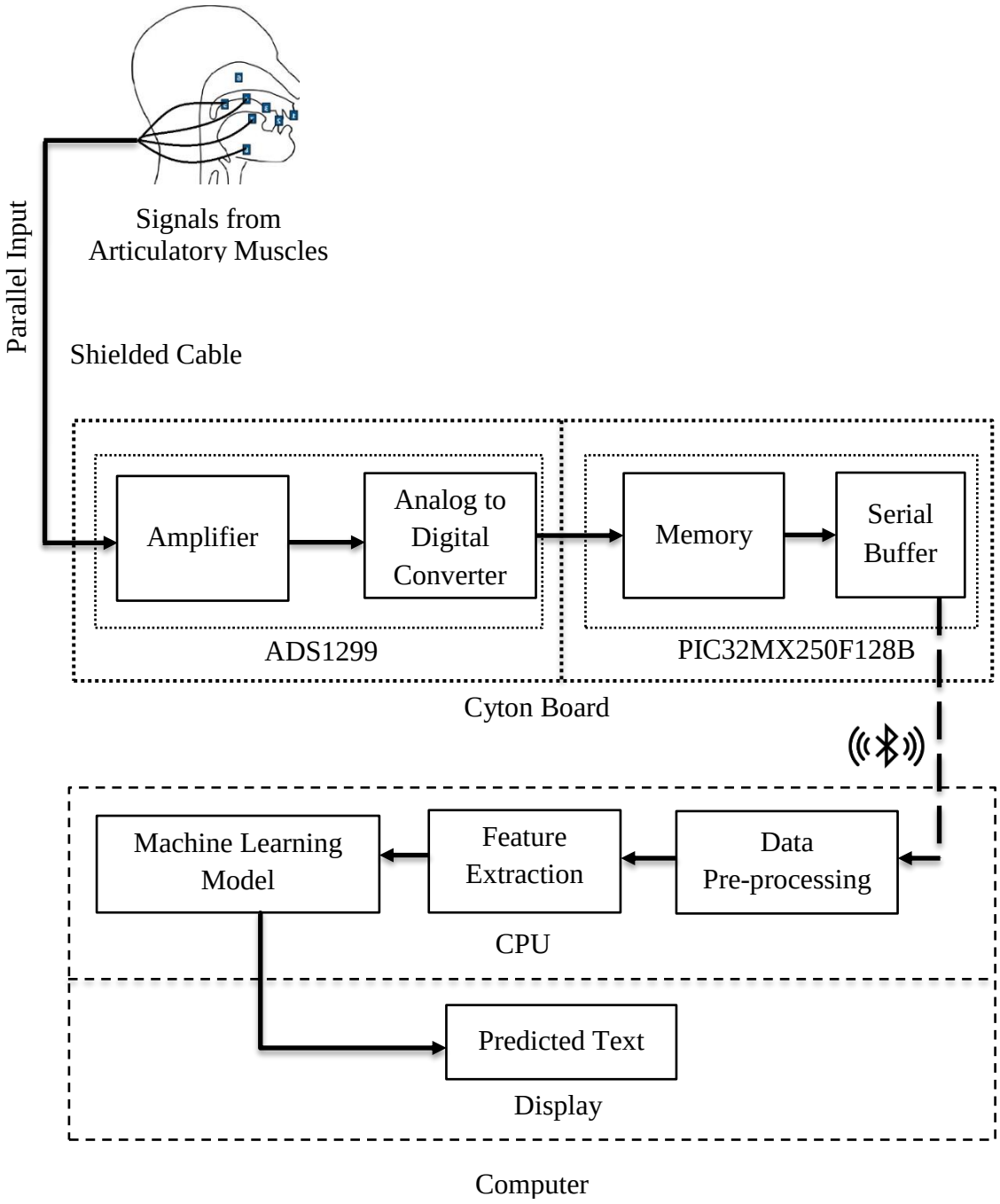

Figure 6-1: System Block Diagram

6.2 Electrode Placement

For the extraction of EMG signals, the electrode placement on the muscles are as shown in the figure below.

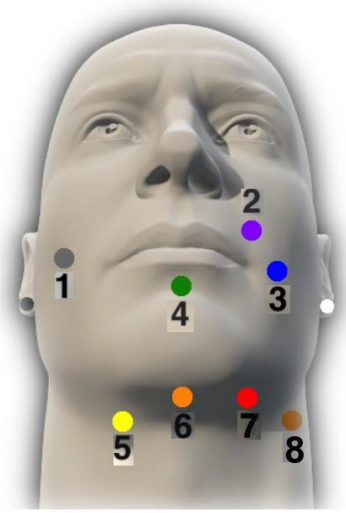

Figure 6-2: Placement of Electrodes

The placement of electrodes is based on the information from the table below. The signals from the electrodes placed on the respective muscles were extracted and fed to Cyton board.

Table 6-1: Electrode and Their Respective Muscle

EMG Channel Number	Muscle's Name
1	Levator Angulis Oris
2	Zygomaticus Minor
3	Zygomaticus Major
4	Orbicularis Oris
5	Omohyoid
6	Anterior Belly of Digastric
7	Mylohyoid
8	Platysma

6.3 Signal Amplification

The extracted analog EMG signal from the electrodes of all 8 channels is fed into respective differential amplifiers which eliminates any common mode noise and amplifies the signals according to the gain of the PGA (Programmable Gain Amplifier) which was set at 24 from the OpenBCI GUI.

6.4 Signal Digitization

For ease of analysis, the analog EMG signal is converted into digital format. The differential output of the PGA is digitized using the 24 bit sigma-delta ADC within ADS1299. The signals are sampled at a frequency of 250 Hz.

6.5 Serial Communication

The PIC microcontroller is the master device for ADS1299 and the Bluetooth module. It interfaces ADS1299 through SPI protocol and the bluetooth module through UART. PIC microcontroller is responsible for configuration of ADS1299, which includes setting the gain of PGA, SPI data transmission rate, sampling rate of ADC and also controls the bluetooth data transmission rate and connections. The PIC microcontroller is heavily involved in fetching digitized data from ADS1299 and sending the data to the Bluetooth module for wireless transmission.

6.6 Wireless Communication

The wirelessly transmitted data from the Cyton board is received by the bluetooth receiver embedded on the USB dongle. Then the USB-to-TTL converter on the USB dongle sends the data from the bluetooth module to the computer through USB.

The Cyton board and the PC send each other some special character from their respective Bluetooth modules. Connection is established after the exchange of special characters. When streaming or recording the EMG data, the data is sent in a packet of 33 byte. The packet contains sample counter, 8 channel data, accelerometer axes values and a footer.

Table 6-2: Data Format

Header	EXG Data	Aux Data	Footer
Byte : 1-2	Byte : 3-26	Byte : 27-33	Byte : 33

The Aux Data consist of the accelerometer and other data. The information about parsing the Aux data is set by the footer. The value of footer determines the value present in it. Normally it consists of the accelerometer data but other settings are also possible.

6.6 Signal Processing

The digitized signal needs to be further processed before it is fed to a classifying neural network. The received signal may contain missing or erroneous data points and other inherent noises. Any faulty data points may cause the neural network to fail to generalize leading to a higher error in classification. The essence of processing the signal is further illustrated in the following literature.

6.6.1 Digital Signal Filtering

The digital signal consists of different types of noises and artifacts which need to be removed. Since the prominent internally articulated signals fall within the range of 1.5-50 Hz a Butterworth filter of order 1 is implemented. The line noise encountered at 50 Hz along with its harmonics at 100 Hz and 150 Hz are suppressed repeating thrice a notice filter of order 1.

6.6.2 Signal Smoothing

Signal smoothing helps eliminate the erroneous data points, some inherent noises and also help extrapolate missing data points in a signal. It does so by incorporating the previous data point and considers that the consecutive data point is representative of the previous data point. It is a type of mathematical convolution and typically implemented on a single dimensional signal model. It generates a time series data constructed by taking averages of several sequential data points of another time series. Moving average algorithms namely Simple Moving Average, Exponential Moving Average, Weighted

Moving Average and Cumulative Moving Average are applied to achieve signal smoothing.

6.7 Extraction of Signal Features

After the EMG signals are processed, they need to be further translated to features that are the actual data contained in a multi-channel EMG signal. Since the EMG signal is very different from the speech signal, it is necessary to explore feature extraction methods that are suitable for EMG to text conversion. Many techniques can be followed for extracting features suitable for audio signals which can be carried over to the EMG signal also and these methods can be described as follows:

6.7.1 Temporal Features Extraction

Temporal features are the time domain features calculated over a window of fixed size that traverses over all the samples in the time domain. Typically, for a sEMG signal, window size of greater than 100 ms and less than 250 ms is used [25] but in cases such as in this project, the EMG signals somewhat follow auditory temporal properties and thus features like Zero Crossing Rate, High Frequency Signals, Rectified High Frequency Signals, Frame Based Power and Double Nine Point Average with a window size of 10ms to 60 ms can be used [25]. From the mentioned features, zero crossing rate, average rectified value, average power and root mean square were selected as they showed significant improvement in the performance of the Neural Network.

6.7.2 Spectral Features Extraction

Spectral features are frequency domain features representing signal amplitude or power against frequency. Spectral features in signal analysis are more helpful as the frequency components are much easier to analyze, add or remove in frequency domain in contrast to time domain.

6.7.2.1 Short Time Fourier Transform

When FFT is used to analyze the frequency domain of a signal, the signal is biased on a finite set of data. When the number of periods is not an integer, the end-points of the FFT become discontinuous due to sharp transitions. These discontinuities show up in the FFT as high-frequency components, sometimes much greater than the Nyquist frequency, which are not present in the original signal. Thus, the spectrum of the signal will be smeared due to the spectral leakage. This causes fine spectral lines to spread into wider signals. Windowing reduces the amplitude of discontinuities at the boundaries of each finite sequence. It multiplies the time record by a finite-length window with an amplitude that varies smoothly and tapers towards zero at the edges. This results in continuous waveform without sharp transitions.

Furthermore, large signals are difficult to analyze statistically as statistical calculations require all the points to be available for analysis. So small subsets of the whole data are analyzed through the process of windowing. It splits the input signal into sufficiently small segments such that the properties of the signal are time-invariant within that segment. It reduces the time domain information and thus resolution in the frequency domain is reduced which implies that there is reduced leakage of spectrum. Thus, before extracting any features in the frequency domain, the time variant data is windowed. It alters the spectral properties of the signal, but the change is designed such that its effect on signal statistics is minimized. All the data points outside the window is truncated while the cut-off points at the ends of the sample will introduce high-frequency components [26]. Based on the different mathematical implementations, windows may be of various types such as Rectangular, Hamming, Blackman, Flattop and Gaussian and so on.

Time-frequency analysis of a signal is typically required to characterize the non-stationary phenomena of signals. The frequency components can be revealed by Fourier transform in chunks of data using sliding windowing technique which is known as Short Time Fourier Transform (STFT). Each transformed complex chunk is added to the matrix which records the magnitude and phase for each point in time and frequency but in doing so all the time related information will be lost. Due to this significant shortcoming of STFT, it is not desirable for wide-band and ultra-wide-band signals, where low spectrogram resolution is observed. However, the selection of appropriate window size

for narrow-band signals such as audio signals, EMG signals, etc. can ideally ensure that the input signal falling within the window remains stationary. But use of very small window size cannot localize the frequency domain. For wide-band signals constant Q transform (CQT) can be used which gives a frequency resolution that depends on the geometrically spaced center frequencies of the analysis window.

Table 6-3: Table of Extracted Features

Temporal Features	**Spectral Features**
Average Rectified Value	Short Time Fourier Transform (STFT)
High Frequency Signals	
Double Nine Point Average	
Frame Based Power	
Zero Crossing Rate	

6.8 Machine Learning Models

After the extraction of features, they need to be fed to a recognition model which classifies the features to their corresponding word (or letter) labels.

6.8.1 Multi-Layer Perceptron

For fast prototyping and verifying the credibility of the extracted features, a simple Artificial Neural Network (ANN) stands to be a very viable option for most machine learning projects. Multilayer perceptron (MLP) is a class of feedforward ANN that learns a function f(X): $R^m \rightarrow R^n$ by training on a dataset, where m is the input dimension and n is the output dimension. Given a set of features $X = (x_1, x_2, ..., x_m)$ and target y, it can learn a non-linear approximator for either classification or regression. It consists of at least three layers of nodes: an input layer, a hidden layer and an output layer with each neuron with linear or non-linear activation function. MLP usually means fully connected network i.e. each neuron in one layer is connected to every neuron on the next neuron.

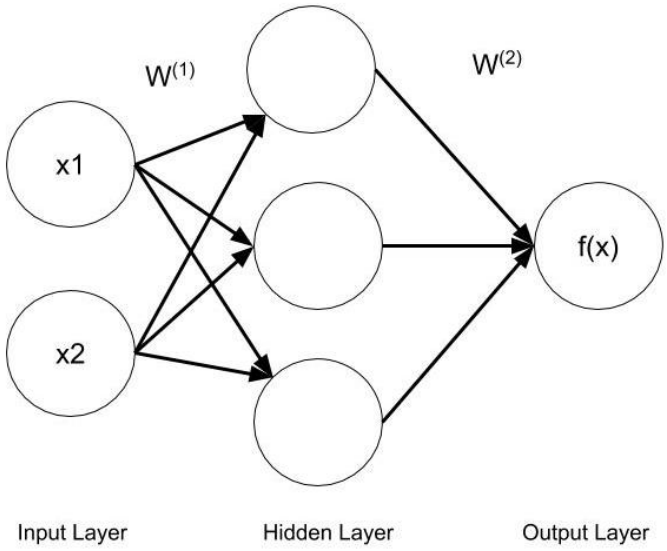

Figure 6-3: A Simple Three Layer MLP Network

Let f(x) be the output vector of the network shown above, $w^{(1)}$ and $w^{(2)}$ be the weight matrices, $b^{(1)}$ and $b^{(2)}$ be the bias vectors and G and s be activation functions then the output of the MLP network shown above can be given by,

$$f(x) = G\left(b^{(2)} + w^{(2)}\left(s\left(b^{(1)} + w^{(1)}x\right)\right)\right) \quad 6.1$$

$$f(x) = G\left(b^{(2)} + w^{(2)}h(x)\right) \quad 6.2$$

The vector $h(x) = (s(b^{(1)} + w^{(1)}x))$ constitutes the hidden layer.

6.8.2 Convolutional Neural Network

Convolution Neural Network (CNN) is basically a regularized version of a multilayer perceptron. MLPs are more prone to overfitting due to their fully connectedness and thus require regularization. CNNs regularize the data using convolution principle i.e. smaller and simpler patterns are used to assemble complex patterns over a hierarchical pattern of data. If f[n] and g[n] are two discrete time data, then the convolution of these two functions are given by:

$$f[n] * g[n] = \sum_{m \to -\infty}^{\infty} f[m]g[n-m] \quad 6.3$$

The set of smaller data points that is compared to input data is known as kernel. The concatenation of multiple kernels, each kernel assigned to a particular channel of input is known as a filter. The filter always has one dimension higher than that of the kernel. As shown in figure 6-4, during convolution operation, the kernel matrix slides over the input data as per the stride value. If the stride value is 1, the kernel moves by a single column of the input matrix. It then performs dot product within the sub-region of the input data and similarity between them is computed. Highest value of output, activation map is produced where the kernel is most similar with the portion of input that is being compared. Let, I(a) be the input and K(a) is the kernel, their convolution C(t) is mathematically defined as:

$$C(t) = \sum_{a} I(a)K(t-a) \quad 6.4$$

$$C(t) = \sum_{a} I(t-a)K(a) \quad 6.5$$

Now, flipping can be done to get cross-correlation

$$C(t) = \sum_{a} I(t+a)K(a) \quad 6.6$$

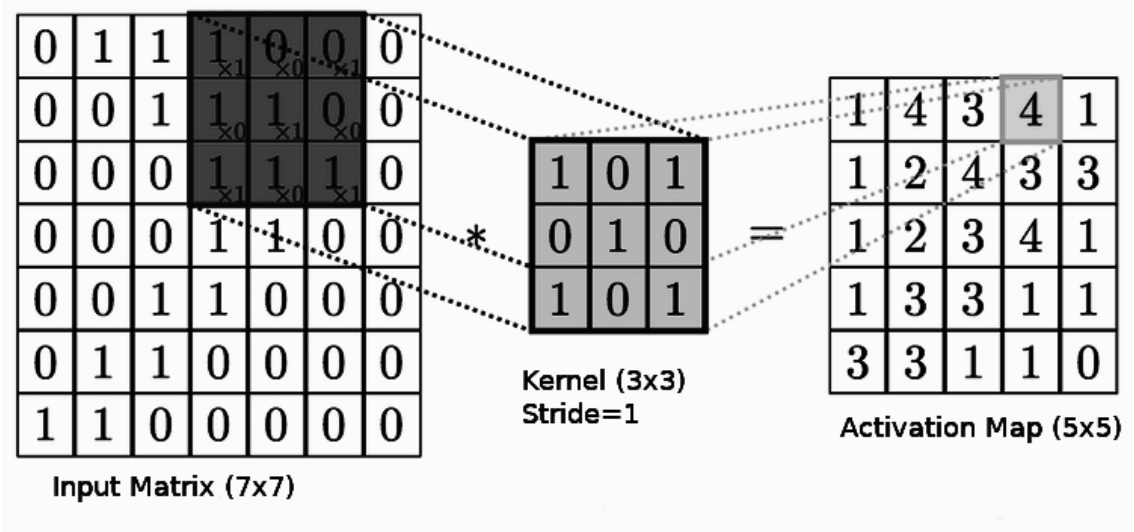

Figure 6-4: Convolution of 7x7 Input Matrix with 3x3 Kernel of Unit Stride

The convolution of these two data represent how the shape of one is modified by the other one. A basic CNN contains a convolution layer, a pooling layer, a fully-connected layer and sometimes dropout layer as well. Convolution layer computes the convolution between the input data based on the above equation 6.3. This layer can also be designed to compute cross-correlation instead of convolution which is basically the same as convolution, the only difference is they are opposite in signs. Cross-correlation being positive is preferred over convolution.

When dealing with high dimensional inputs, it is impractical to connect all the neurons of a layer to every neuron in the previous layer because such a layer wouldn't consider the spatial structure of the data. CNN exploits spatially local correlation by enforcing a sparse local connectivity pattern between adjacent layers which means each neuron is connected to only a small region of the adjacent layer. The extent of this connectivity is determined by the hyperparameters; depth, stride and zero padding. They control the size of the output volume of the convolution layer. Depth of the output volume corresponds to the number of kernels to be used. The stride is the unit by which the kernel is to be slid over the input matrix. Zero padding allows to control the spatial size of the output volume by padding with zeros around the border [27]. The spatial size of the output volume (O) can be computed as,

$$O = \frac{W - K + 2P}{S} + 1 \quad (6.7)$$

Where W = input volume size or input dimension,

K = receptive field size of convolution layer or kernel size,

P = amount of zero padding,

S = stride

A Pooling layer (PL) is an effective way of nonlinear down-sampling. It has kernel size and stride as non-learnable parameters. The size of stride is the same as that of the kernel by default. The exact location of a feature is less important than its relative location with respect to other features. This layer progressively reduces the spatial size of the representation for latter layers, memory footprints and computation complexity of the network but adds no new parameters. It also contributes in controlling overfitting. Due to its destructiveness, PLs are very rarely used or discarded in case of very small dataset. Various pooling layers such as max-pooling, average-pooling, l2-norm pooling, etc. are used in a neural network. [28]

Mathematically, max-pooling function is defined as,

$$p_{i,m} = \max_{1 \le n \le G} q_{i,(m-1)*s+n} \quad (6.8)$$

Where G is pooling size and s is the shift size or stride size

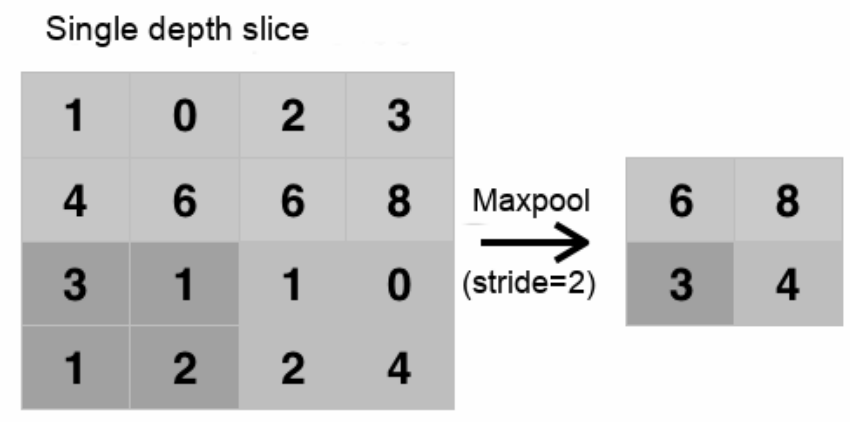

Figure 6-5: Max Pooling of a Slice With a Stride of 2 Units

A fully connected layer is usually a linear layer while a dropout layer is a layer implemented for data regularization which randomly drops out some neurons of the layer on which it is implemented to avoid data overfitting. Neurons have connections to all the activations of the previous layer. Their activations can thus be computed as an affine transformation; preserving parallelism, with matrix multiplication followed by a biased offset.

6.9 Display Unit

The OpenBCI GUI stores recorded data as a text file in a folder. The utterances from the user is recorded and stored at a specific location. These utterances are separated and are passed through the same signal processing and data processing pipeline through which the data for training machine learning model was passed through. This pipeline outputs the features which are further passed to a pre-trained model. The model then predicts the utterance on the basis of the features and the predicted utterance is finally displayed on the display unit of a remote computer.

7. IMPLEMENTATION DETAILS

The hardware and software discussed in the previous chapters explained how each of them worked using theoretical and mathematical approaches. This chapter describes how those systems were put into practice.

7.1 Hardware Implementation

This section elaborates on how each of the components in hardware has been used, what kind of setup has been followed and how the designed circuit has been fabricated.

7.1.1 Self Designed Hardware

During preliminary phases of the project, EMG signals were extracted and preprocessed using self-designed dual channel hardware. It included signal extraction, amplification and filtering. The components were fabricated in a matrix board.

7.1.1.1 Parameter Calculation

The Instrumentation Amplifier AD620 was used as a pre-amplifier for the circuit. The EMG signals are weak and are in presence of common mode signals. So it is necessary to have a good CMRR to amplify the EMG signals and reject strong common mode signals. For EMG signals it is recommended to have CMRR greater than 90 dB. From the Figure 12-1, a gain of 10 provides CMRR greater than 90 dB steady within the frequency range of 1 - 100 Hz which is the working band frequency of the system. And from Figure 12-2, curve of AD620 shows the gain remains steady within the frequency range. The gain of AD620 can be changed by changing the value of R_G. Equation 7.1 gives the relation of R_G and gain of the instrumentation amplifier and the gain was set to 10.8 using a resistor (R_G) of value 5KΩ. [29]

$$R_G = \frac{49.4}{G-1} K\Omega \qquad 7.1$$

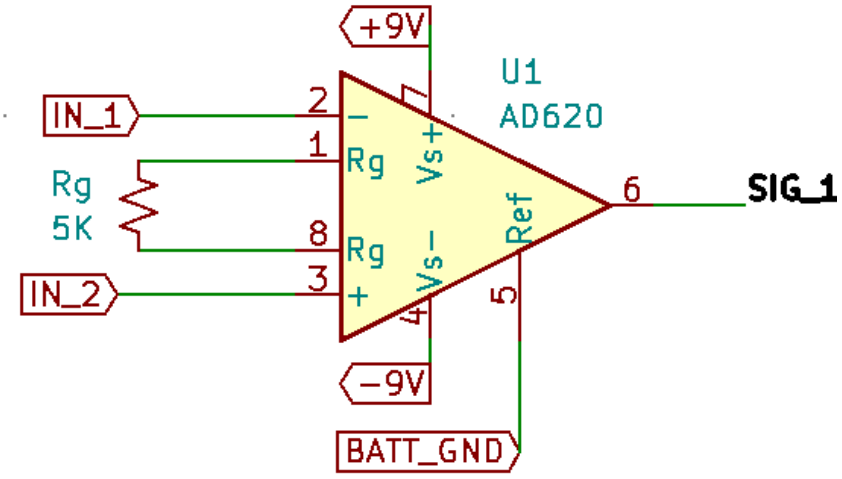

Figure 7-1: Instrumentation Amplifier

The small gain of about 10 was chosen so that noise signals of relatively low amplitude in comparison to the EMG signals do not get amplified to a significant level. This stage cannot avoid noise but on further filtering, the noise with less amplitude are suppressed very well or are more prone to attenuation. And upon choosing gain less than 10 the CMRR might not be sufficient to reject stronger common mode signals and also amplify the necessary signals.

The signal passes through a high pass filter with cutoff frequency of 1 Hz. The equation 4.4 gives the relation of the frequency, resistors and capacitors from which the required values are obtained. But the Instrumentation Amplifier gives bipolar output so electrolytic capacitors are avoided which have the higher capacitance value instead ceramic capacitor is used which is non-polar. The ceramic capacitor has less value so a high value of resistor is chosen.

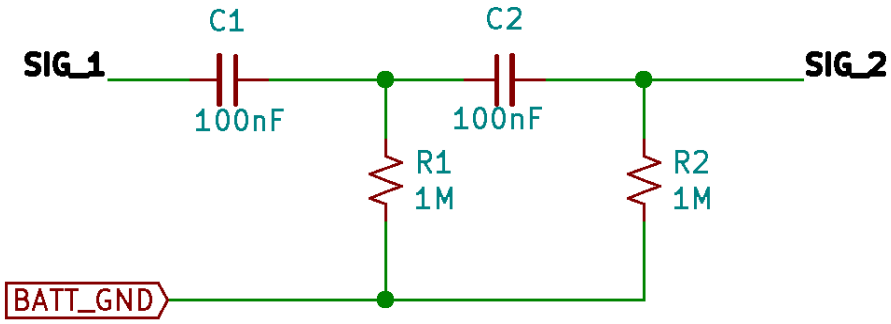

Figure 7-2: High Pass Filter

The signal from the high pass filter is then amplified by the non-inverting amplifier. The gain of the amplifier is controlled by the resistor R1 as shown below. The signal from the high pass filter is very low and with much experimentation the value of R1 was set to 470 KΩ resulting in a gain of about 471.

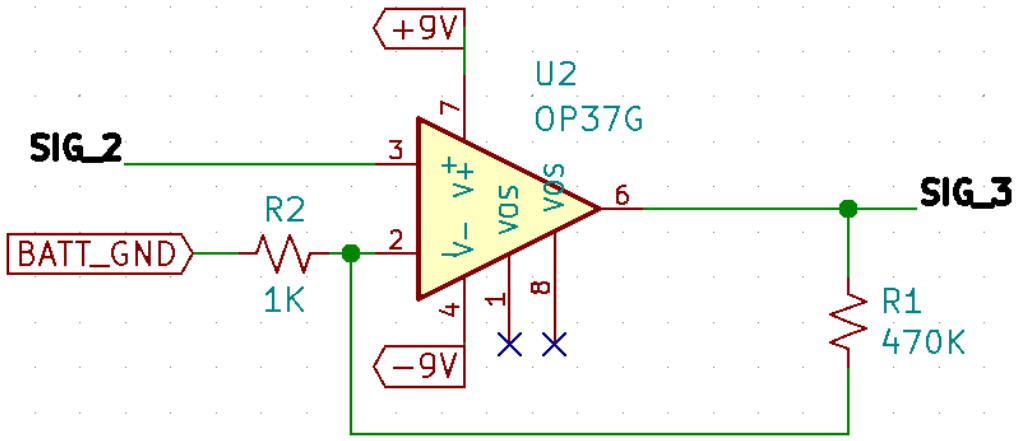

Figure 7-3: Amplifier

To create a band pass of 1 - 100 Hz, a low pass filter is added. The cut off frequency of the low pass filter is 100 Hz. It is an active low pass filter with a gain of 1.5, which is the value of A_0. The gain is adjusted using the equation 4.8 which gives the relation of R4, R3 and A_0. The value of Q becomes 0.667 with the selected value, which is less than the value referenced at equation 4.13. The calculated values are tabulated in the Table 7-1.

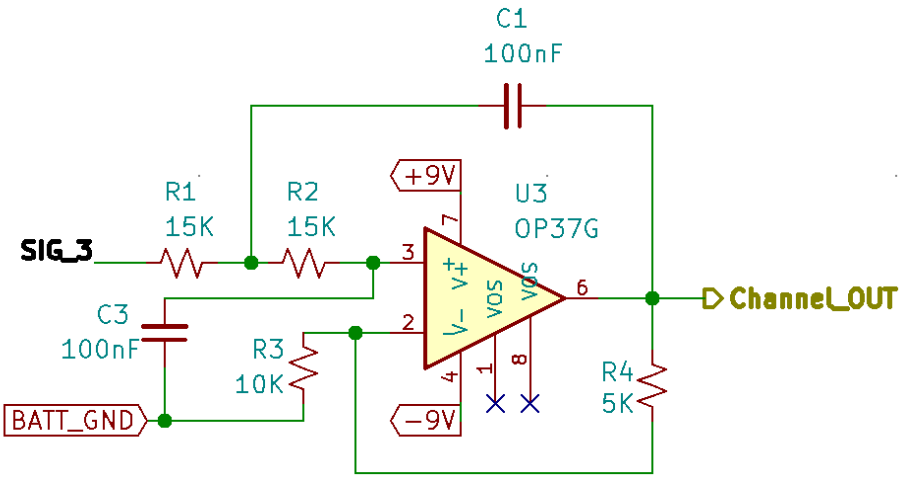

Figure 7-4: Low Pass Filter

The calculated values are tabulated below with the respective circuit above.

Table 7-1: Circuit Parameter Calculation

Circuit Element	Resistor Values (KΩ)	Capacitor Values (nF)	Cut-off (Hz)	Gain
IA	Rg = 5	-	-	10.8
HPF	R1 = 1000 R2 = 1000	C1 = 100 C2 = 100	1	1
Amplifier	R1 = 470 R2 = 1	-	-	471
LPF	R1 = 15 R2 = 15 R3 = 10 R4 = 5	C1 = 100 C2 = 100	100	1.5
Total Gain				7630.2

7.1.1.2 Schematic and Layout Design

Schematic is a simple representation of a circuit design on a two-dimensional plane that shows the functionality and connectivity between the different components in the circuit. Utilizing the schematic, CAD software knows the inter-connection between the components, the types of components used and how the components have been used.

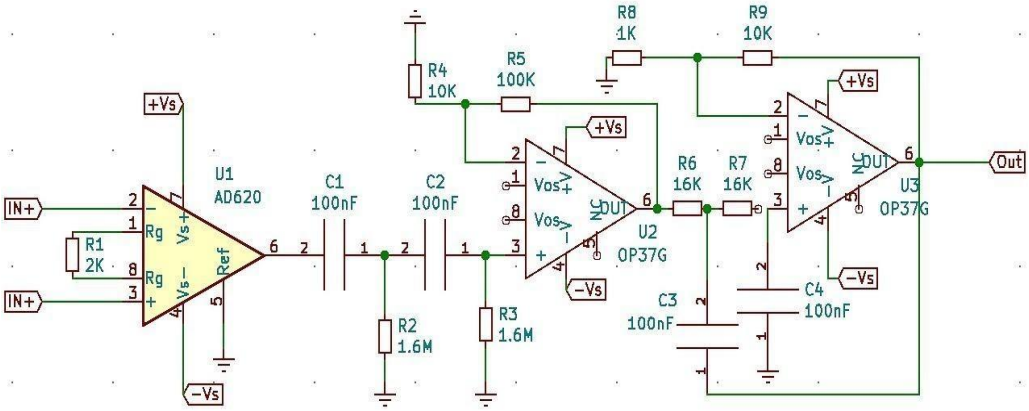

Figure 7-5: Branch Sheet of Designed Schematic

A multi-channel EMG data acquisition schematic has redundant circuit elements and circuit connections that makes schematic design unconventionally complicated and difficult to follow. To counter the redundancy and to keep the schematic design conventional, hierarchical sheet schematic design was followed. The multi-channel repetitive circuit elements were kept on a branch sheet that was referenced in the root sheet whenever it was needed as shown in figure 7-6.

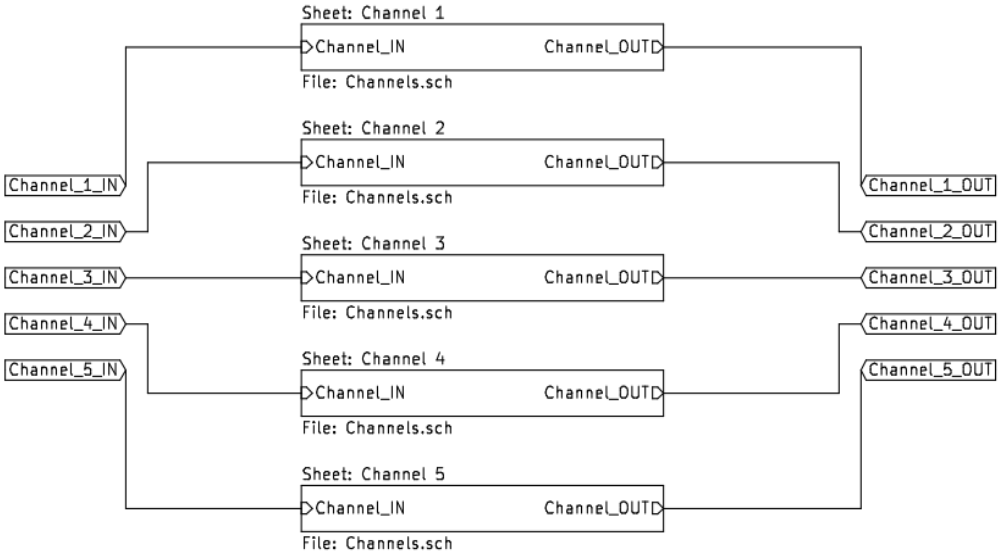

Figure 7-6: Root Sheet of Designed Schematic

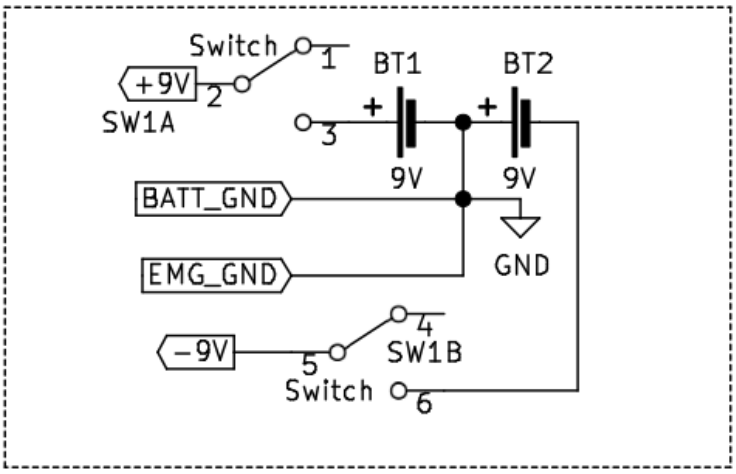

Figure 7-7: Schematic of Power Supply

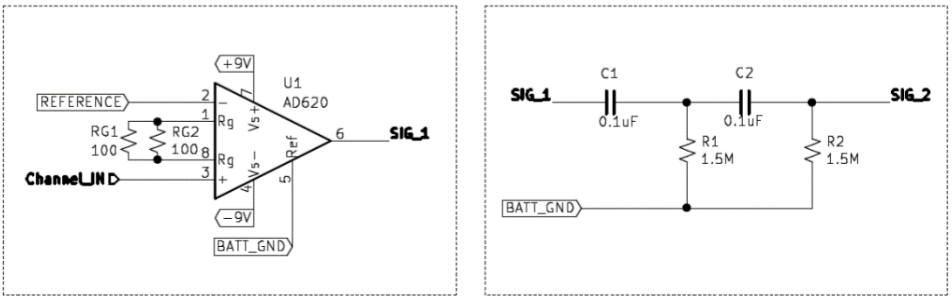

Figure 7-8: Instrumentation Amplifier and High Pass Filter Block

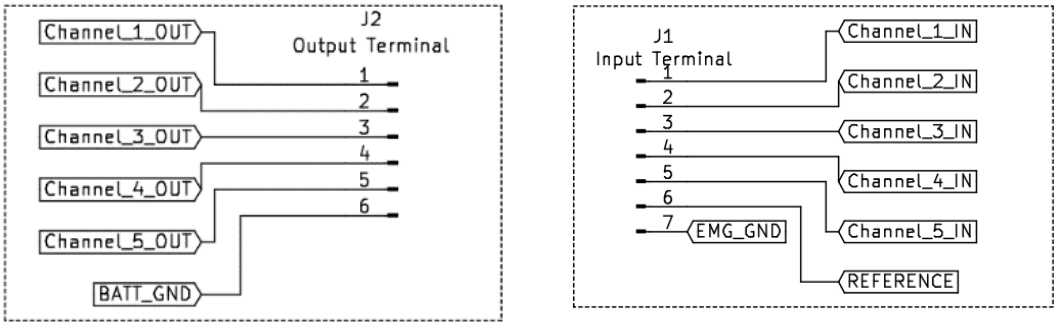

Figure 7-9: Output Terminals (Left) and Input Terminals (Right)

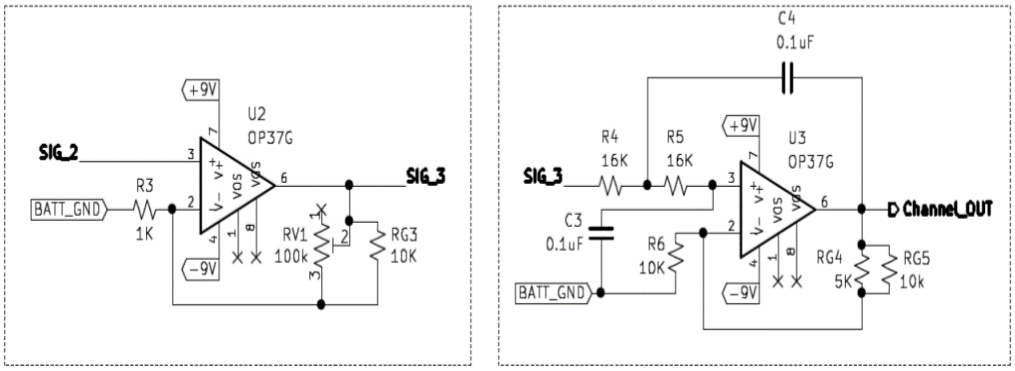

Figure 7-10: Amplifier (Left) and Low Pass Filter (Right) Block

Layout design refers to design of the printed circuit board taking in consideration all the aspects due to which the functioning of the system gets hindered. The layout design can be as important as the circuit design to the overall performance of the final system. The signals involved in a circuit could be analog or digital. It is advised to always isolate such circuitries to avoid interference between the two. The EMG acquisition hardware too has both the analog and digital part. The instrumentation amplifier circuit, the filter circuit and the amplifier circuit belong to the analog part whereas the Arduino belongs to the digital part. The designed PCB only hosts the analog circuits and the Arduino is kept separate with only necessary wires interconnecting them. Both the analog and digital part of the circuit is battery powered to avoid any AC coupling in the circuits.

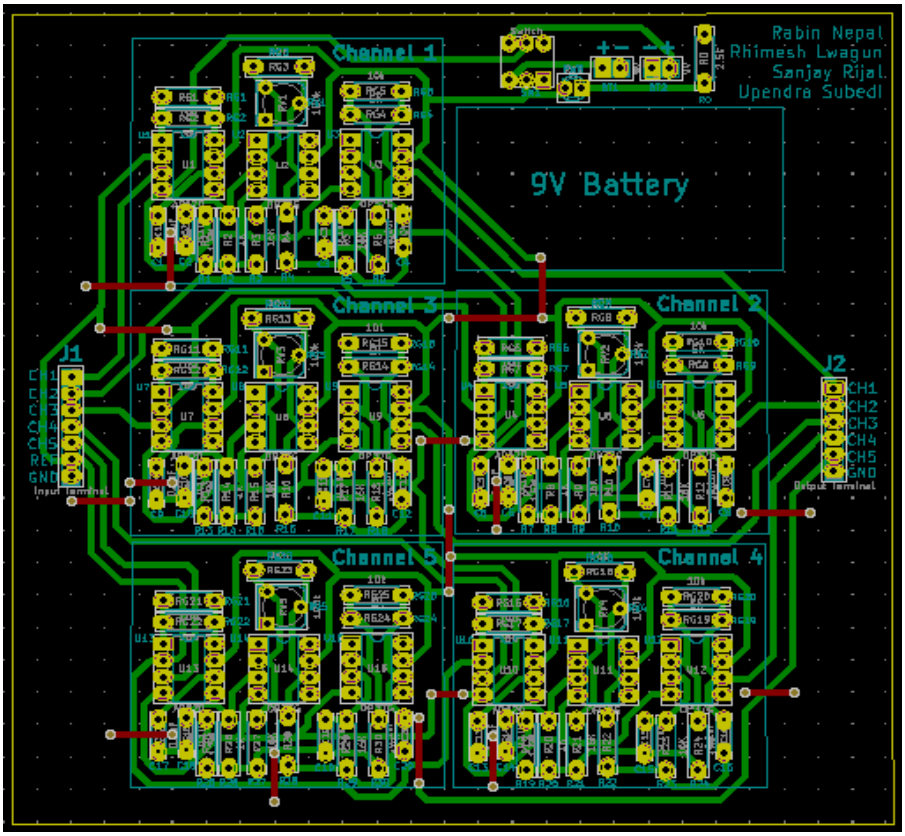

Figure 7-11: PCB Layout

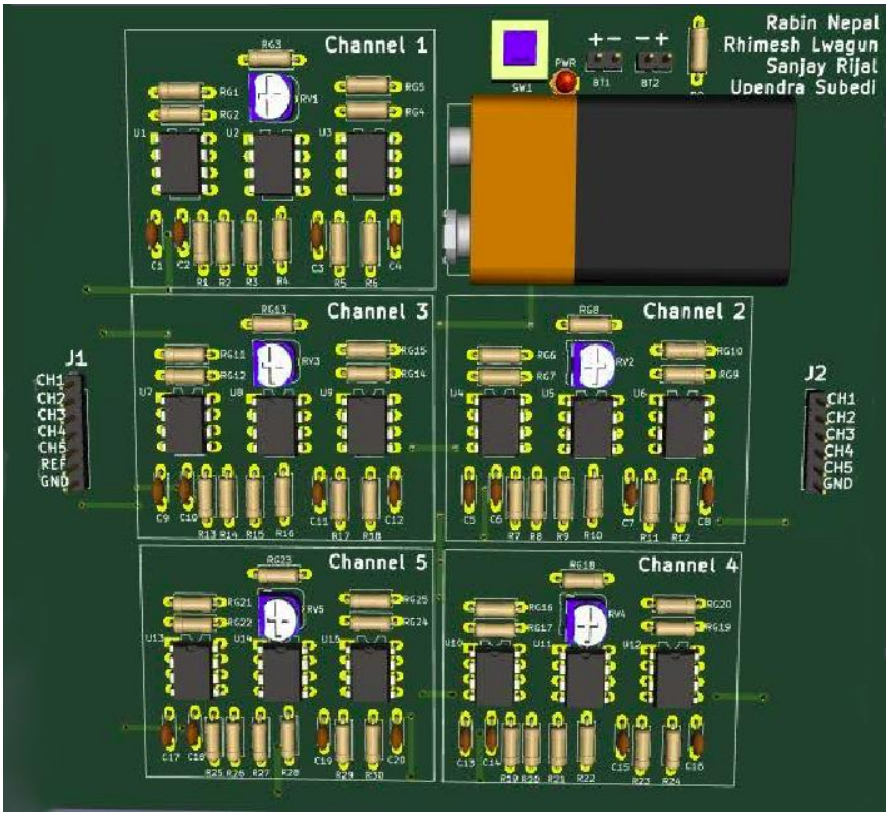

Figure 7-12: 3D View of Designed PCB

The copper trace width for the circuit is restricted to 1mm (40 mils) by the fabrication process involved. Any traces below 1 mm gets eaten away by the chemical solution along with the other unmasked copper regions. The fabrication process also restricts the design to a single sided PCB as the exact alignments of different layers in a PCB is difficult without computerized machineries. However, with available resources a fully functioning circuit was designed. The 3D view of the designed PCB layout is shown in figure 7-12.

7.1.1.3 Hardware Operation

The amplifier circuit and the filter circuit was initially tested individually by giving an input of low frequency signal from a function generator and the output was visualized in an oscilloscope. The gain of the amplifiers and the cut-off frequency of the filters were inspected during the test. The circuits were then tested with EMG signals from facial muscles extracted using the Ag-AgCl electrodes in bipolar mode. The signal electrodes E1 and E2 were placed on the cheek muscle (Zygomaticus Major) and the ground electrode G1 was placed on the wrist as shown in figure 7-13. The user was then asked to twitch his cheek muscles and the EMG signals were observed in an oscilloscope.

Figure 7-13: Initial Setup for Circuit Testing

After successfully testing the individual circuits of the EMG acquisition system, a throughout system with a tunable gain was designed for two channels as shown in the figure 7-13. The circuit was tested with bipolar signals from two articulatory muscles;

Zygomaticus Major (Channel 1) and Platysma (Channel 2), as shown in figure 7-8. The signal electrodes E1 and E2 for Channel 1 were attached along the length of the muscle Zygomaticus Major keeping them at a certain distance apart. Similarly, the signal electrodes E1 and E2 for Channel 2 were attached across the length of the muscle Platysma with some distance in between them to avoid cross-talk. The ground electrode G for both the channels was placed on the wrist away from the signal electrodes.

The overall circuit gain was first set at 5000 and was gradually increased until the signals of proper amplitude were observed which was at about a gain of 7600. The signals were sampled with Arduino's ADC at a sampling rate of about 600 Hz. The signals were then recorded for a discrete word utterance for an interval of 3 seconds, which was the average time for the user to utter a word.

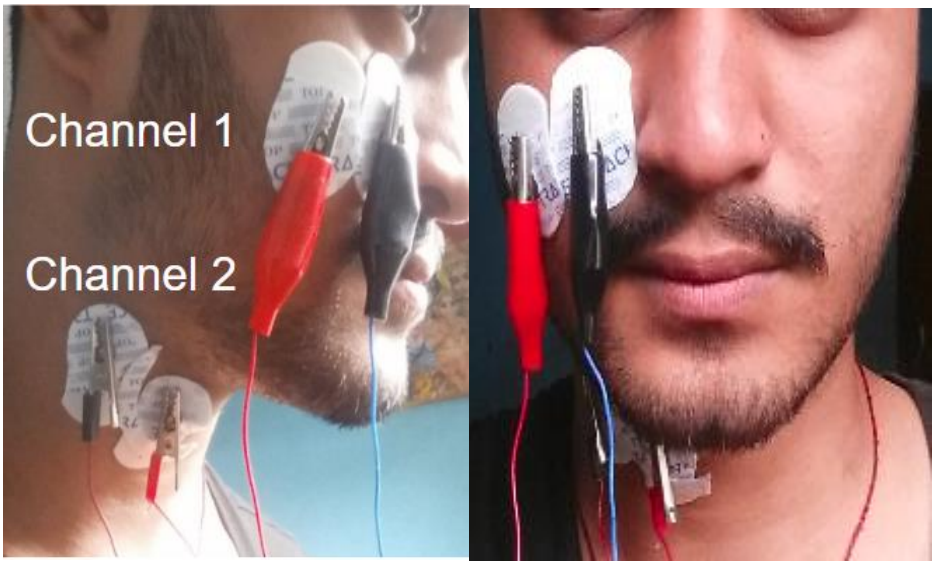

Figure 7-14: Electrode Placement on Facial Muscles

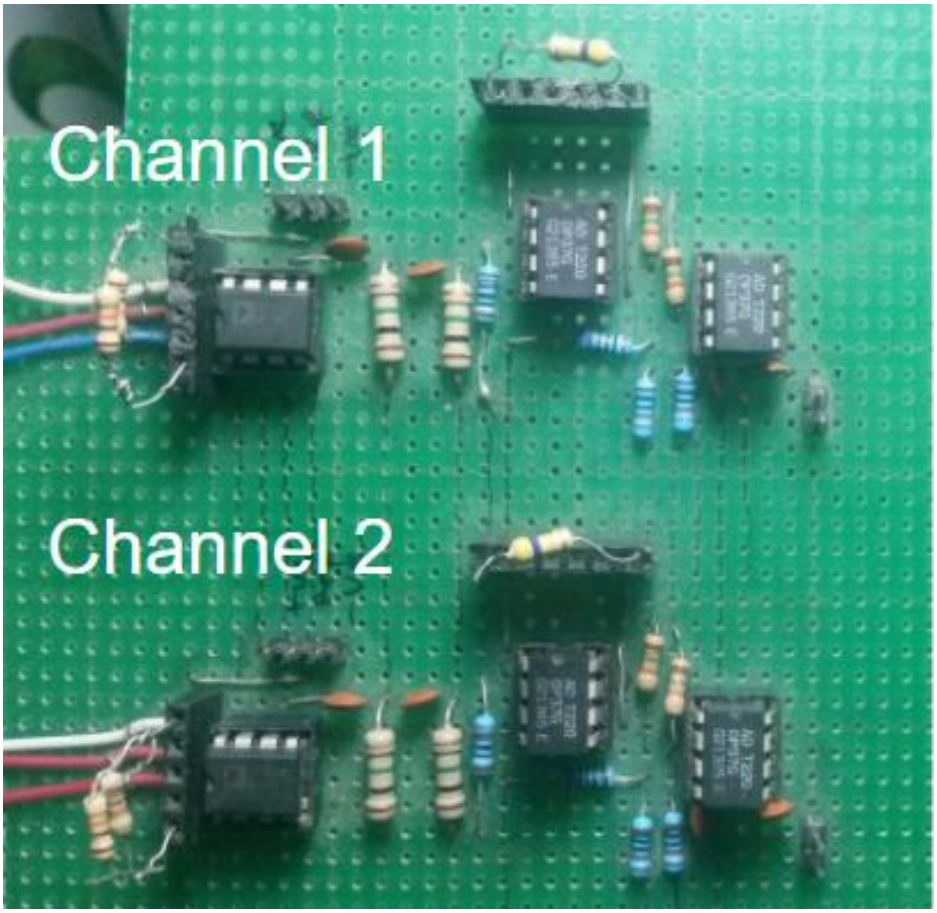

Figure 7-15: Dual Channel EMG Acquisition Circuit

The above figure shows the designed circuit implemented on a matrix board. The upper section is designated as channel 1 and the lower section is designated as channel 2. The pair of three wires going into both the sections are signal wires E1 and E2, and ground wire G1. The ground wire is shorted to circuit ground and the signal wires are fed to the differential input of the instrumentation amplifier AD620. The circuit has female headers in both sections for making the resistors swappable so that the gain of the system can be tuned accordingly. The filtered EMG signals are sent to Arduino using jumper connectors through the male headers present at the end of the circuit.

7.1.2 OpenBCI Cyton Board

Cyton board integrates hardware components to extract the weak biosignals. The connection leads with the electrodes at the end needs to be connected to the Cyton board and to the target muscles properly to extract the EMG signals. The board also needs to be configured to accurately convey information from the acquisition hardware to the remote computer.

7.1.2.1 Leads Connection

The EMG signals are extracted in two ways namely monopolar and bipolar. In bipolar both the differential electrode is placed on the muscle of interest. In monopolar configuration one of the electrodes from the differential electrode is placed as reference and another on the muscle of interest. Both configurations involve placing the ground electrode on the ground of the body (bony or muscular part of the body).

OpenBCI, Cyton board is used to collect the EMG signals in monopolar configuration. Although it can be deployed to collect in bipolar configuration, the number of electrodes on the face is gradually reduced and more muscles can be monitored in monopolar configuration. The Cyton board consists of 8 channels, ground and reference (labelled as SBR2). The bottom pins are used to connect to the array of the gold electrode cups. The ground and reference are connected to the ear lobe on both sides, which can be interchanged. The remaining electrodes are connected to the muscle of interest. The gold electrode cups are filled with the conductive paste. Then they are attached to the surface of the skin locating the muscles**.** The array of the electrode is connected to the pins in the Cyton board accordingly.

7.1.2.2 ADS1299 to PIC connection

The array of the gold electrodes are connected to the pins of Cyton boards which connects to the ADS1299. The ADS1299 amplifies each weak EMG signal with the gain of 24. The signals are then sampled at 250 Hz by 24 bit ADC. The ADS1299 and PIC are connected by SPI bus. The signal is then sent to the PIC. The configuration of ADS1299 is done by the PIC through the same bus. The PIC comes with a pre-installed firmware,

developed by OpenBCI community. The PIC sends the received signal to the bluetooth module through UART.

7.1.2.3 PIC and Bluetooth Integration

The Cyton board with its bluetooth module sends data wirelessly to the remote computer possessing the USB dongle. This makes the Cyton board portable and less prone to possible noises. The PIC sends the received sampled data from the ADS1299 to bluetooth module through UART. The data is then sent to the PC in a packet. The USB dongle receives the packet sent by the Cyton board and parse it accordingly.

7.2 Software Implementation

This section explains about acquisition and processing of data, preparation of dataset and information regarding the deployment of machine learning model.

7.2.1 Data Acquisition

This portion briefly explains the mechanisms regarding the acquisition of data along with its manipulation using designed hardware which was later cascaded with Arduino.

7.2.1.1 Data collection from Self Designed Hardware

For data collection and training of the system a python script was written. It records Serial data for a certain fixed time. The serial data from the Arduino is stored to a CSV file under a suitable labelling. The columns in the CSV file indicates the channel and rows are the data of the respective channels.

For further convenience a GUI interface capable of visualizing and recording was developed. But the features are still under development. It is able to visualize the raw EMG signal from the Arduino. It has two threads, one for receiving the serial data and another for the update of the plot. The features FFT, save and record emits signal when triggered and respective event is performed.

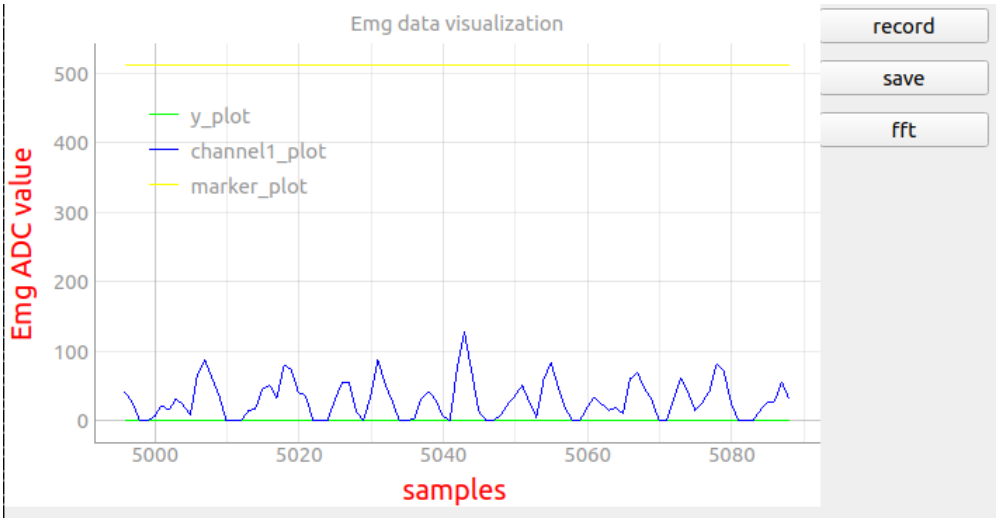

Figure 7-16: Custom Graphical User Interface

The y-axis gives the raw EMG ADC value from Arduino. The x-axis gives the number of samples. The line y-plot is used for testing and debugging the plot. The channel1 plot indicates the signal from the Arduino, number of EMG can be increased and set different colors for other plots. The marker plot is the ending value sent by the Arduino, which usually contains 1024, 512 or any value. As this value also helps in setting the maximum range of the graph plot.

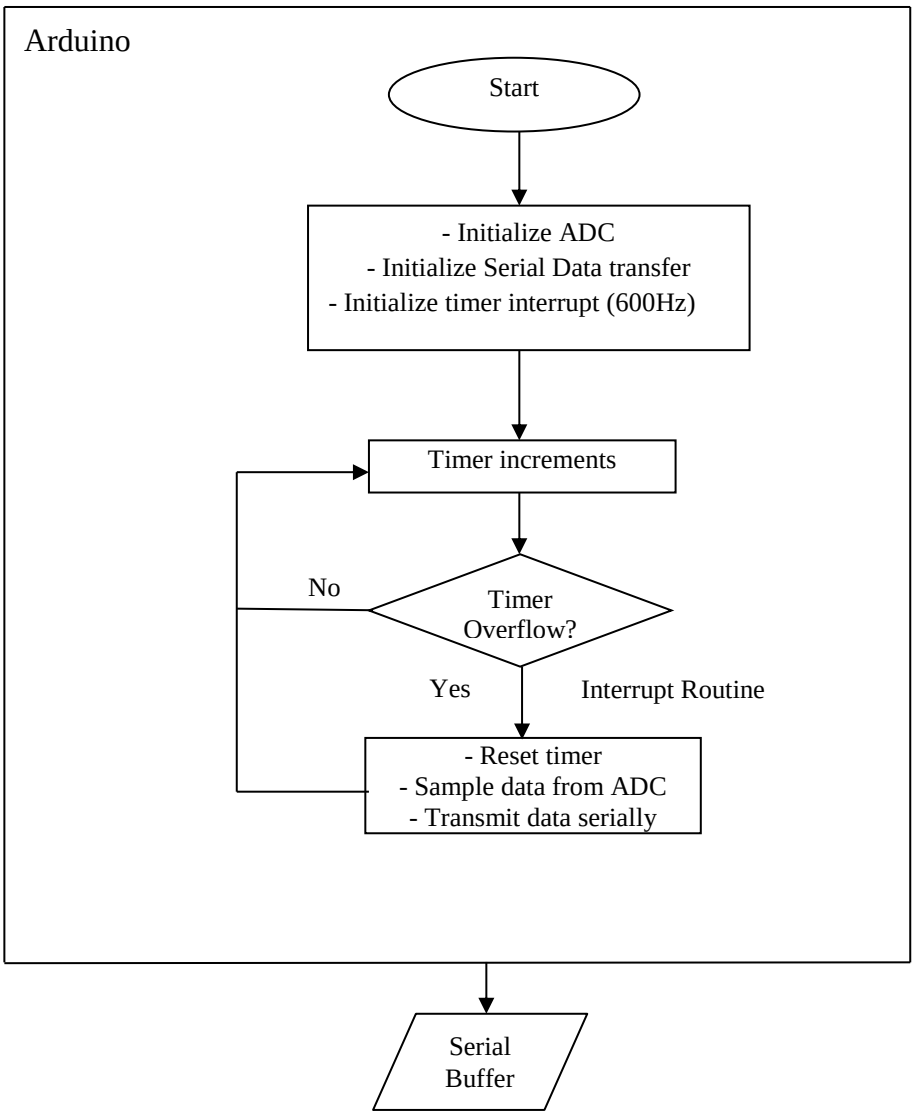

Figure 7-17: Flowchart of Serial Data Transfer from Arduino

The Arduino and the computer work in unison to record the EMG signal. The program execution and interaction of the Arduino and computer can be realized from the following flowchart. The Arduino samples data with frequency of 600 Hz which is set with the timer which is then transmitted to the computer through serial interface. The computer intermittently checks the availability of data in the serial buffer and continues to record the data till the given recording time period.

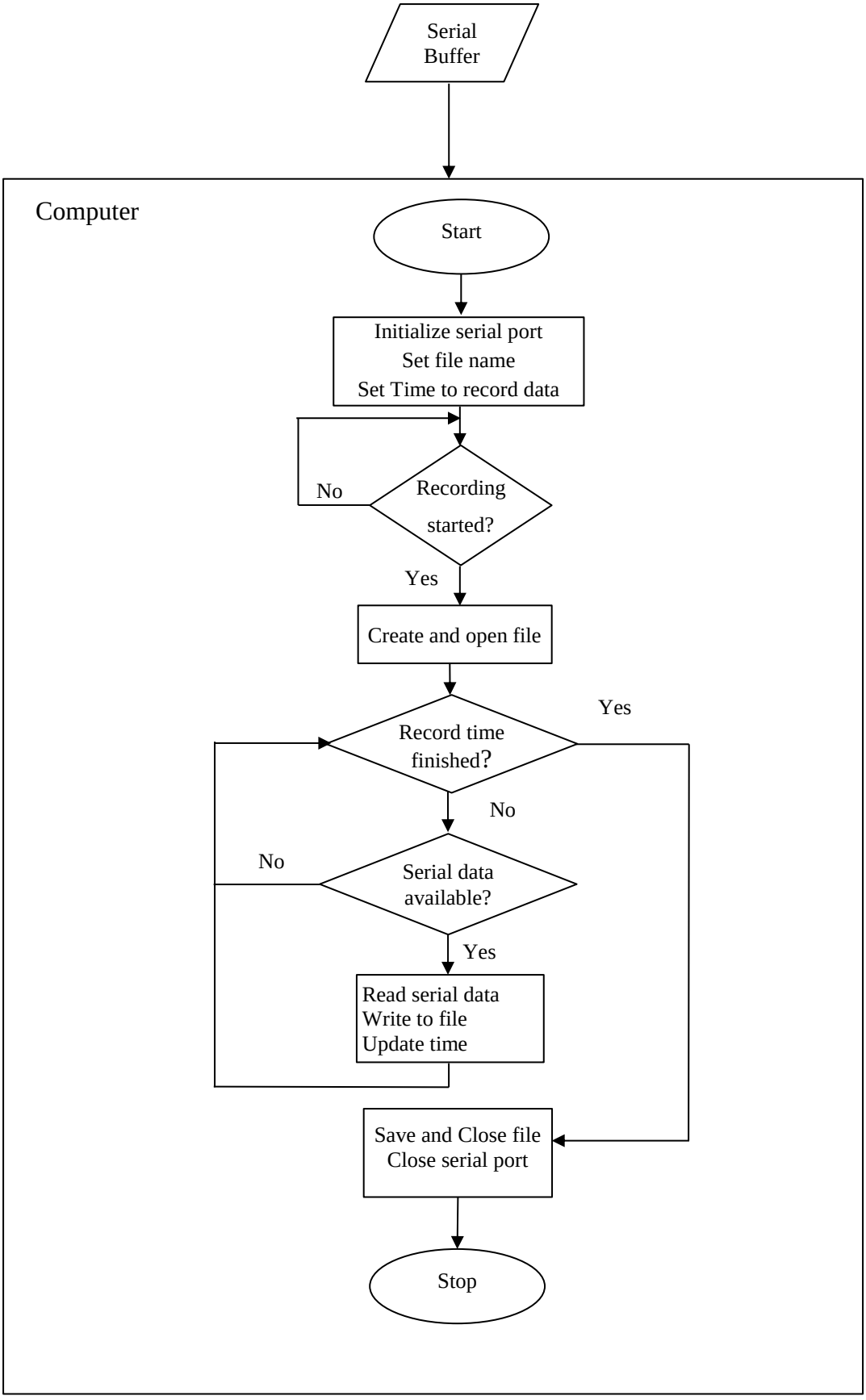

Figure 7-18: Flowchart of Serial Data Receiver in Computer

7.2.1.2 Data Collection from OpenBCI Hardware

The OpenBCI Cyton board allows for easier recording of EMG signals from the muscles and provides a GUI for controlling, visualization and storing data. The extracted data was sent wirelessly to a remote computer running the OpenBCI software. The GUI of the software is shown in the figure below:

Figure 7-19: OpenBCI GUI

Data from each channel was visualized in distinct color in the GUI and are being saved simultaneously once the session was started. The software stores not only the EMG signals from the Cyton board but also the time information, digital pin states, analog pin data and accelerometer data of all axes.

The Cyton board has an onboard push button associated with the pin D17 of the PIC32 microcontroller which can be pressed while recording the data. Utterance instances of the words were represented by an enabled D17 pin whose value was also stored alongside with the signal in a text file. So the state of the D17 pin helps to differentiate samples of the uttered words from the recorded session file. On separating the samples with help of D17 pins, digitized signals were obtained.

7.2.2 Signal Processing

The raw signals stored in a text file from the OpenBCI software are passed through multiple signal processing steps to obtain a noise free signal within the desired frequency band of 1.5 Hz to 50 Hz. The line noise and heartbeat artifacts encountered needs to be removed and the DC drift in signal amplitude should be corrected.

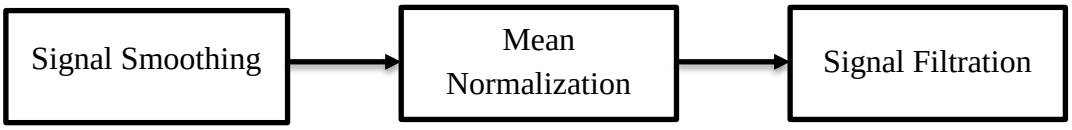

Figure 7-20: Signal Processing Block Diagram

7.2.2.1 Signal Smoothing

The moving average filter provides resistance to the sudden change in amplitude in the signal that usually occurs due to random noises superimposed with the signal. The raw signal is smoothed using the 8 point moving average filter which averages out the data points so that the random variation in amplitude can be omitted. A time domain plot of the raw signal vs the moving averaged signal can be seen in the figure 7-21. It can be observed from the figure that the spurious signals are filtered out.

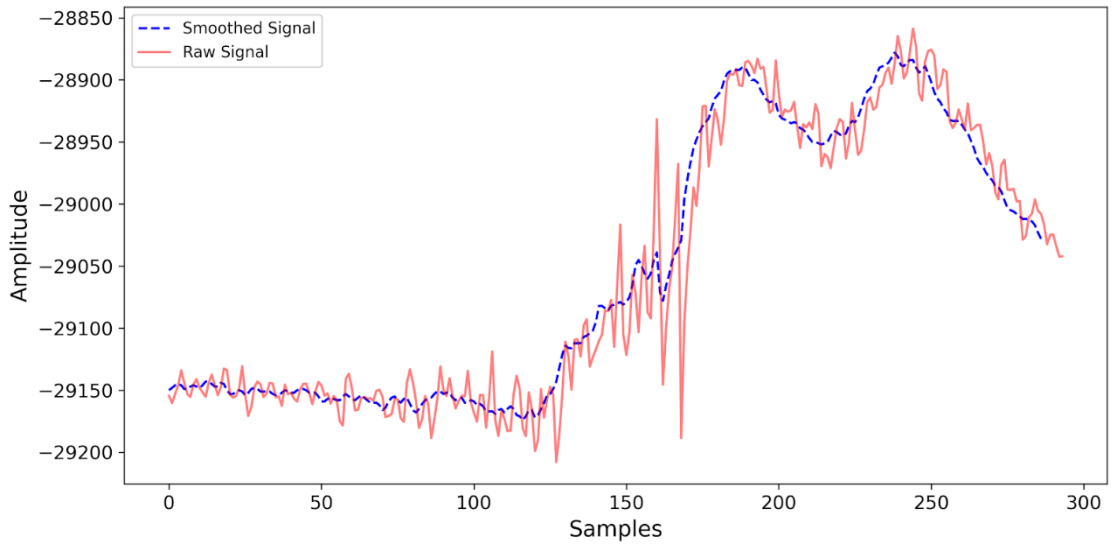

Figure 7-21: 8 Point Moving Averaged Signal

7.2.2.2 Mean Normalization

Due to the DC offset drift, the signals tend to diverge from the baseline. The DC offset occurs due to the variation in electrode placement relative to the reference electrode. Also the OpenBCI hardware introduces a very low frequency source of noise when recording potential difference between signal and reference electrodes. These effects are prominent on longer recordings. The rate and direction of drift is unpredictable and is different to each channel. It is corrected by mean normalization of the signal.

7.2.2.3 Signal Filtering

The digitized data needs to be filtered before further processing as it consists of signals of unwanted frequency range, ECG artifacts, and line noises along with their harmonics. The ECG artifacts are removed by using Ricker Wavelet Transform. The line noise and its harmonics is suppressed by a digital notch filter at 50 Hz, 100 Hz and 150 Hz. The signal is further filtered by a high pass and low pass Butterworth digital filter of first order at a cut-off frequency of 1.5Hz and 50 Hz respectively. Despite the application of a low pass filter at 50 Hz, a notch filter is still used to eliminate the line noise as the roll-off of the first order Butterworth filter is quite low and fails to significantly attenuate the line noise at 50 Hz.

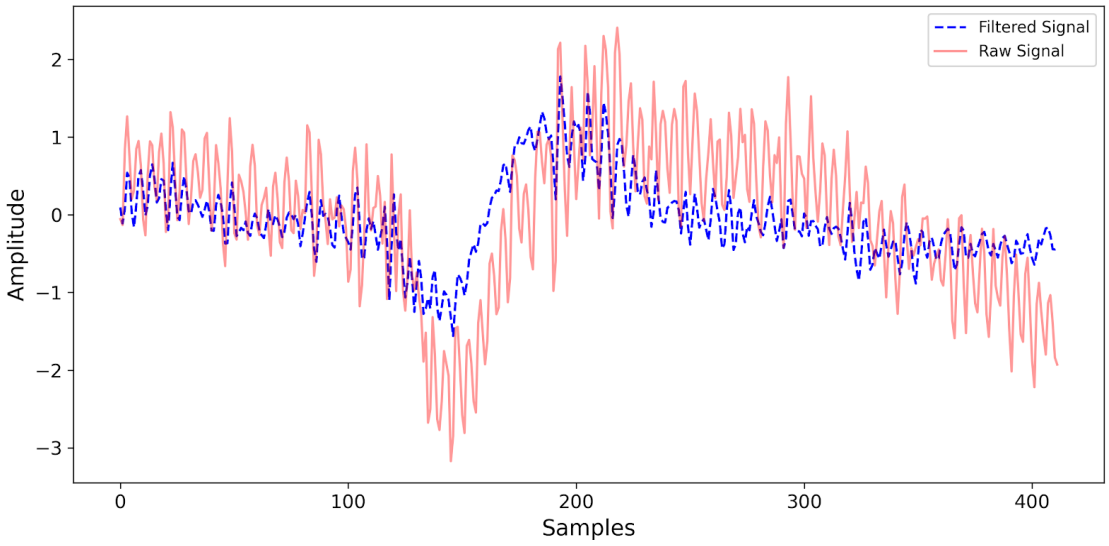

Figure 7-22: Filtered and Raw Signal (Time Domain)

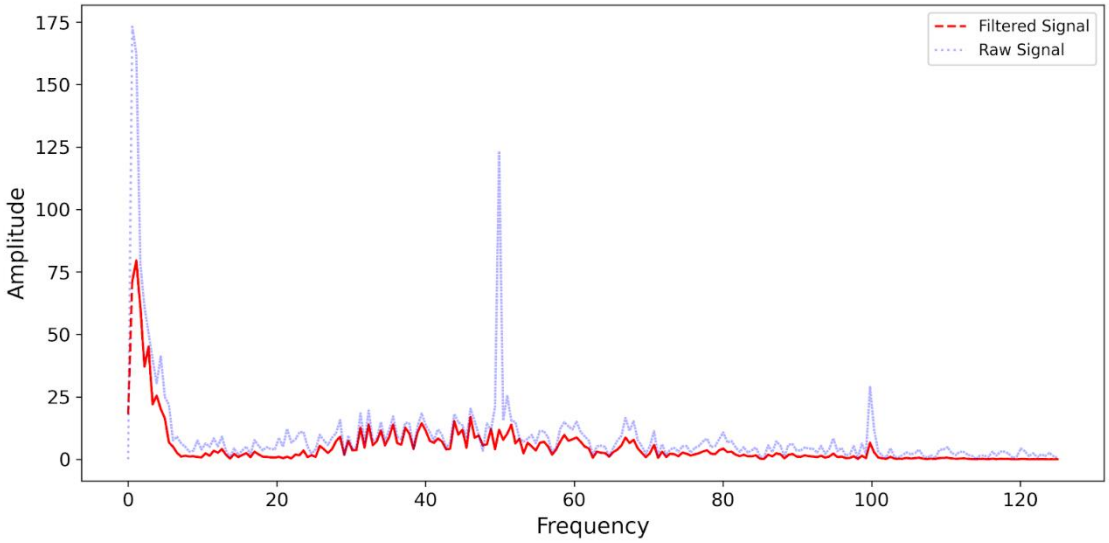

Figure 7-23: Filtered and Raw Signal (Frequency Domain)

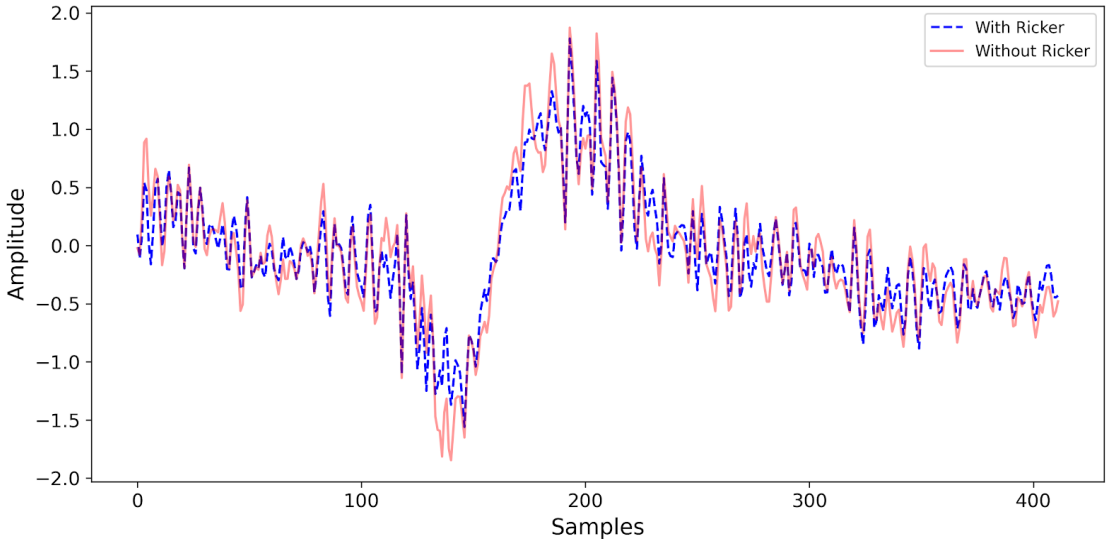

Figure 7-24: Ricker Wavelet Filtered Signal (Time Domain)

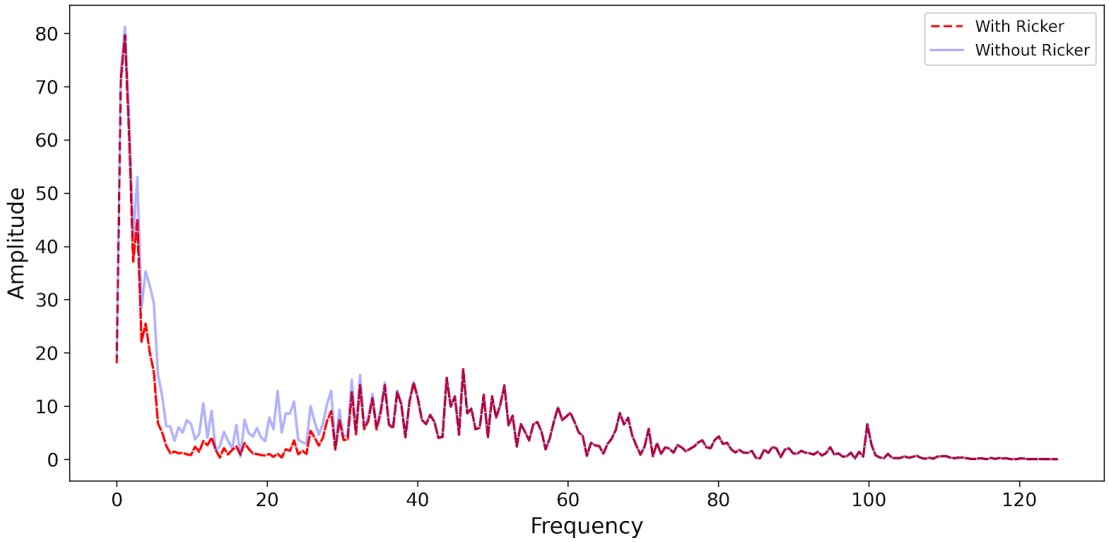

Figure 7-25: Ricker Wavelet Filtered Signal (Frequency Domain)

7.2.3 Data Processing

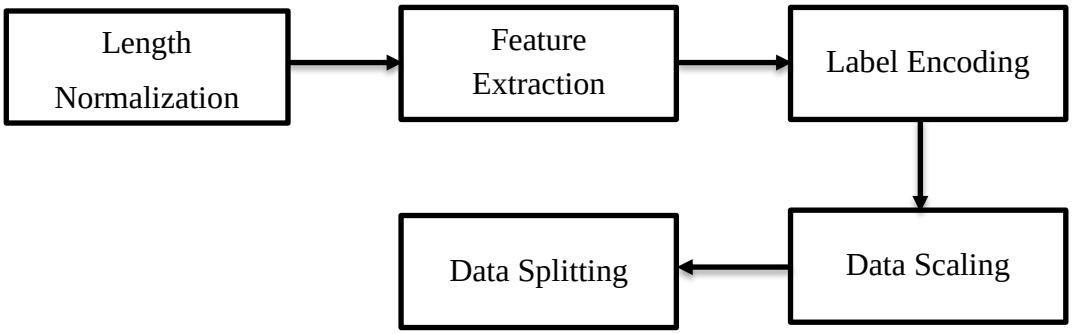

Figure 7-26: Flow of Data Processing Before Model Training

7.2.3.1 Length Normalization

During the analysis of data, it was observed that length of data samples varies slightly for utterance to utterance of a word for a speaker and greatly from speaker to speaker. This difference in sample length was normalized by dropping the data instances with length greater than the 900 samples and zero padding the data instances with smaller sample length.

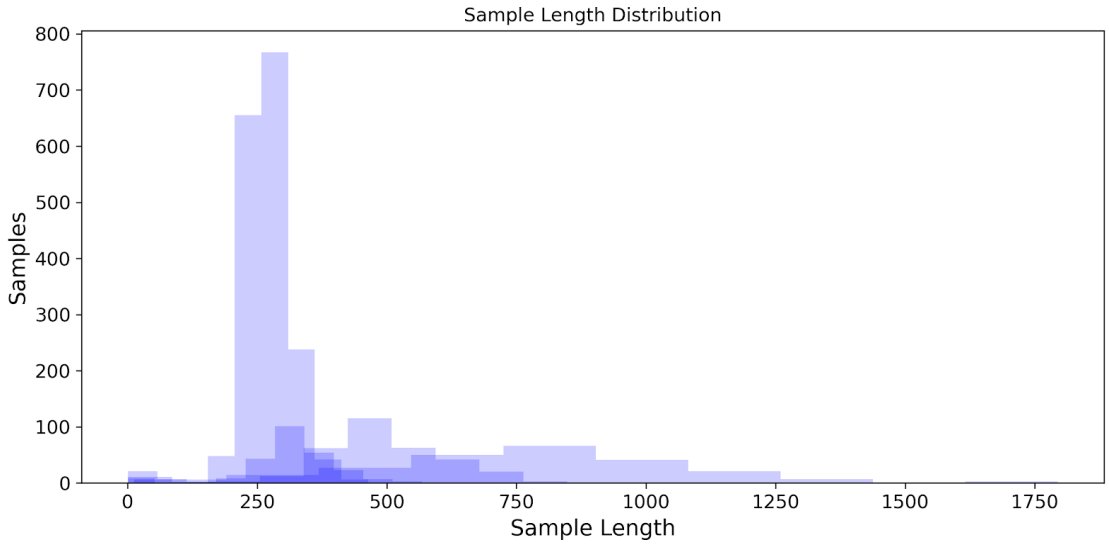

Figure 7-27: Distribution of Sample Length Before Normalization

The cut-off length was selected so that most of the data was included and only the outliers were left out. To determine the cut-off length, a list containing the length of all the samples was generated and the 95th percentile of this list was calculated. This was repeated for each speaker in both the speaking modes and finally an average of all the 95th percentile was taken to incorporate data instances of all speakers in all modes. The data instances beyond the 95th percentile were ignored and the rest were zero padded to this value.

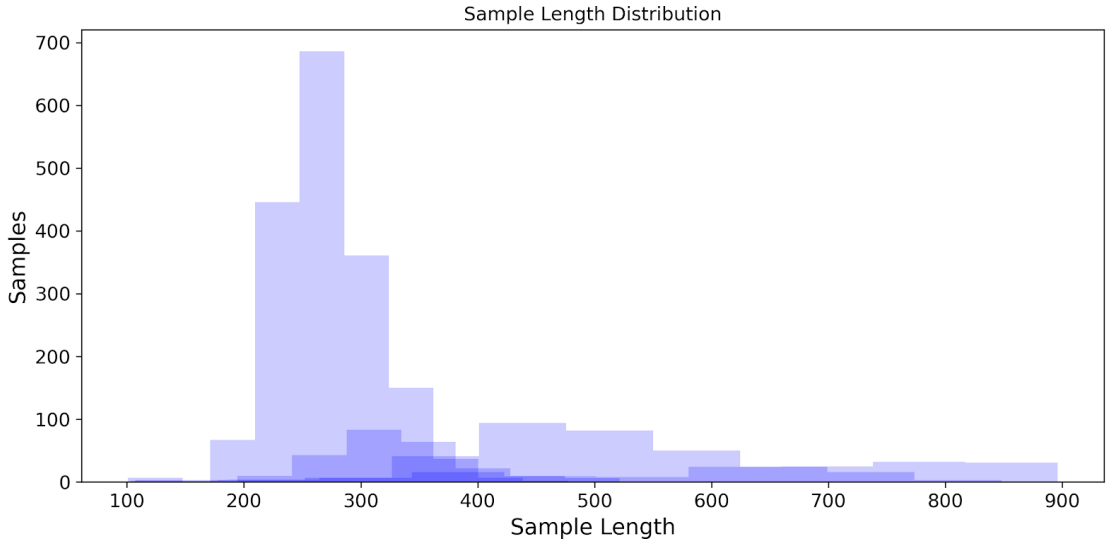

Figure 7-28: Distribution of Sample Length After Normalization

7.2.3.2 Feature Extraction

The filtered signals do represent the utterance more than the raw signals. But the features from the signal are also able to impart more information to the model so the temporal as well as spectral features of the data were extracted. Under temporal features, zero crossing rate, nine point double average, high frequency signal and frame based power were selected as they showed significant improvement in the performance of the Neural Network. Double nine point average of a signal was plotted against the raw signal as shown in figure below. It shows that the feature double nine point average mainly focuses on the unvarying nature of the signal.

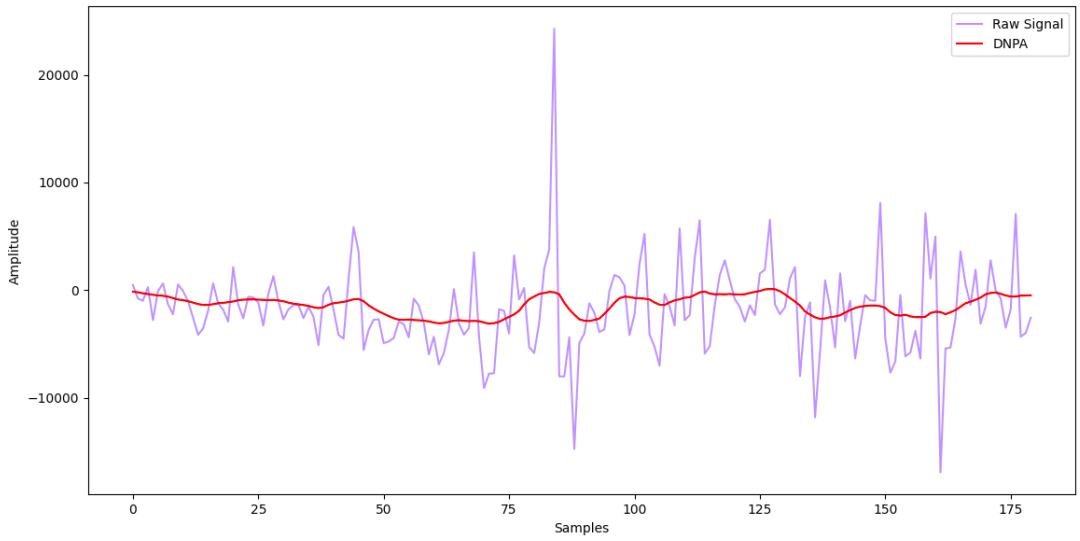

Figure 7-29: Raw and DNPA Signal Plot

The features high frequency signal and rectified high frequency signal has been plotted along with the corresponding raw signal in the figure below. These features emphasize on the dynamic quality of the signal.

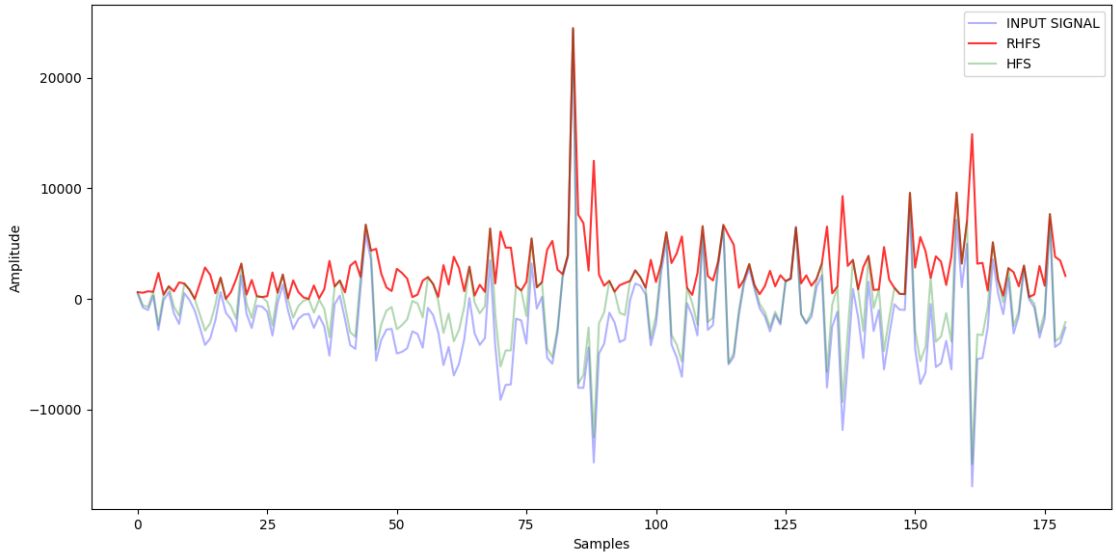

Figure 7-30: Raw, RHFS and HFS Plot

STFT was implemented to extract the spectral features. The output of STFT were frequency samples, time segments and complex transform features. The complex transform features were 3 dimensional tensors which were reshaped into 2 dimensional tensors before feeding into the 1-dimensional input layers of MLP and CNN. The STFT of the raw signal was plotted which is as shown in the figure below. The raw signals have a dominant amplitude near 200, 400 and 600 samples which is also observable in its STFT plot. The signals with the prominent amplitude are of frequencies less than 2 Hz which is within the range of internally articulated EMG signals.

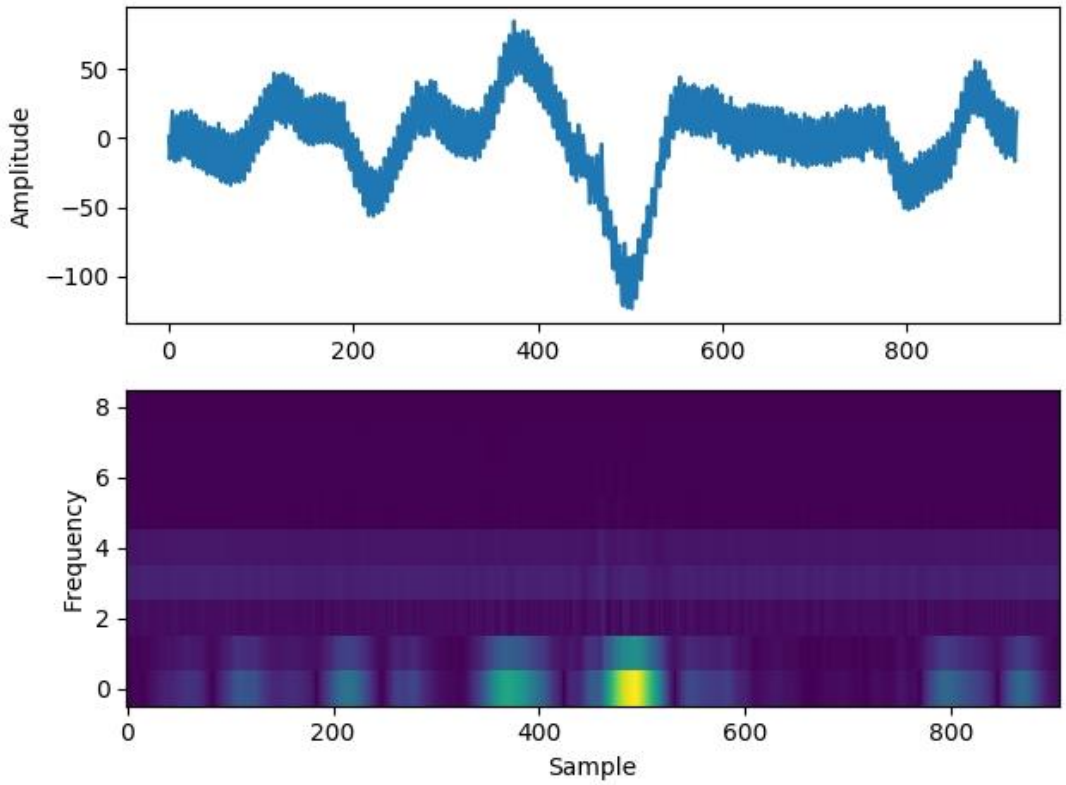

Figure 7-31: Raw Signal and STFT Plot

7.2.3.3 Label Encoding

The class labels are in string format which needs to be encoded into numeric format for classification tasks. The label encoding of the dataset was achieved using Sklearn's LabelEncoder method.

7.2.3.4 Data Scaling

After encoding discrete words, the data was scaled to transform all the data to a common scale without distorting differences in the ranges of values or losing any information. Firstly, data from all the channels were converted to microvolts (μV) and a channel-wise standardization was performed by computing z-scores for all the respective channels. The time and frequency domain features were then extracted and a feature vector was obtained. This feature vector was again scaled for normalizing the different value ranges of different kinds of features. Finally, the mean and variance values were also stored to scale the data during the model deployment stage.

7.2.3.5 Data Splitting

The dataset was split into two subsets; train set and test set. The train set was used to fit the designed model while the test set was used for analyzing the performance of the model. The test set was given input to the model and the predictions made by the model were compared to the theoretical expectation of the model. Loss or cost value is obtained after the comparison which is performed using a loss function. Among various loss functions, sparse cross-entropy or log loss function was used which is a classification loss function and measures the cross-entropy or the differences between two probability distributions. Entropy is the average number of bits required to identify an event drawn from the dataset. A skewed distribution has low entropy and the distribution where events have equal probability has high entropy [30]. The test data was repeatedly analyzed by the loss function to optimize the model.

7.2.4 Machine Learning Model Development

This section describes the architectural structures of machine learning models MLP and CNN. As the size of layers and selection of activation function alters the result, they need to be properly selected for desirable output from the model.

7.2.4.1 Architecture of MLP Model

The architecture of the designed MLP network is as shown in Figure 8-20. It consisted of an input layer with a number of neurons equal to the size of input feature tensor, two hidden layers with ReLU activation functions consecutively and finally an output layer with 10 neurons. The input is a 1D tensor of size 1x200 which was fed to a hidden layer H1 with 64 neurons that gave an output of 1D tensor with a size of 1x64. The next hidden layer H2 shares the same properties as the layer H1. The both hidden layers implemented ReLU activation function. Finally the output of size 1x64 from the hidden layer H2 was mapped to the output layer with 10 neurons using softmax regression.

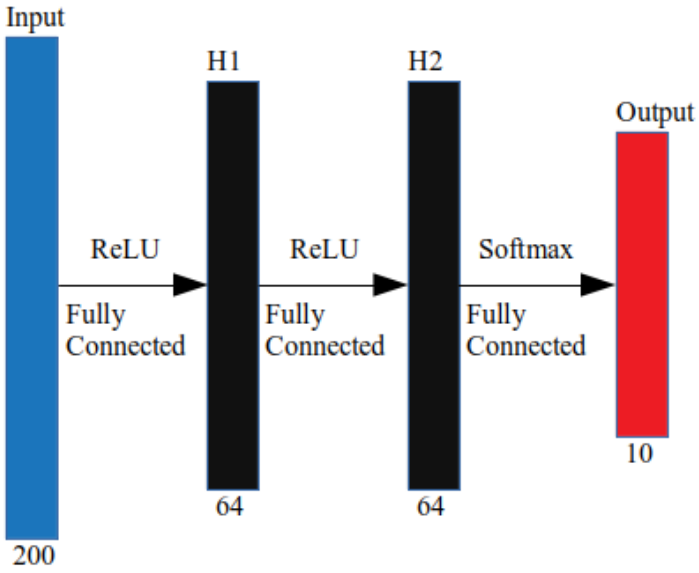

Figure 7-32: Designed Multilayer Perceptron (MLP) Model

7.2.4.2 Architecture of CNN Model

The architecture of designed CNN is as shown in the figure 8-21. The size of the input tensor Number of samples x Number of channels. The first hidden layer was a 1-dimensional convolution layer (Conv1D 1) with ReLU activation function. It consisted of 100 filters and a kernel/filter of size 12. It convoluted the data resulting in an output which was then pooled with 1-dimensional max-pool layer (MaxPool-1D 1) with kernel size of 2. The data was again fed to 2nd Conv1D with the same number of filters and kernel of size 6 with ReLU activation function. This layer was then fed to the 2nd MaxPool-1D layer with the same properties as that of Maxpool-1D 1. The data was then flattened to one dimensional tensor using a Flatten layer. The next hidden layer was H1 with the number of nodes being 100. The final layer was the output layer with a resulting

tensor size of 10 which was mapped from the hidden layer H1 using softmax regression. All the hidden layers along with the output layer beyond the Flatten layer were fully connected.

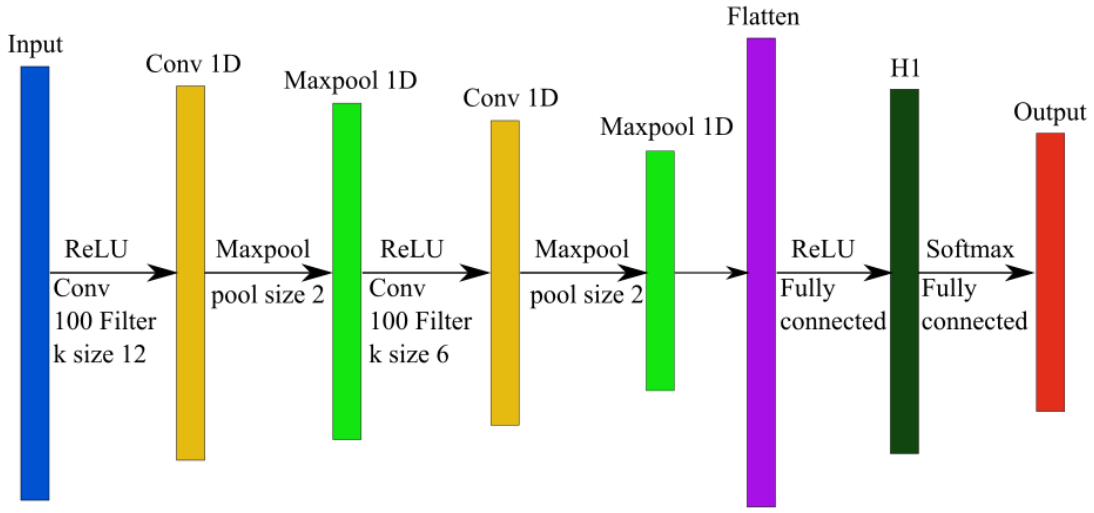

Figure 7-33: Architecture of Designed 1D CNN

8. RESULTS

This section includes responses of circuit and different machine learning models. The extracted EMG signal was visualized in real time and recorded. Machine learning models were tested and relevant outputs are illustrated through figure and plots.

8.1 Circuit Response

This portion pertains to the responses obtained from the self-designed hardware along with that from OpenBCI Cyton board.

8.1.1 Self-Designed Hardware Response

Using a python sketch, the words uttered by a subject was recorded. This data was then manipulated using Octave and FFT was visualized without applying any digital filter. The recorded EMG data of some of the words are as shown in figures below:

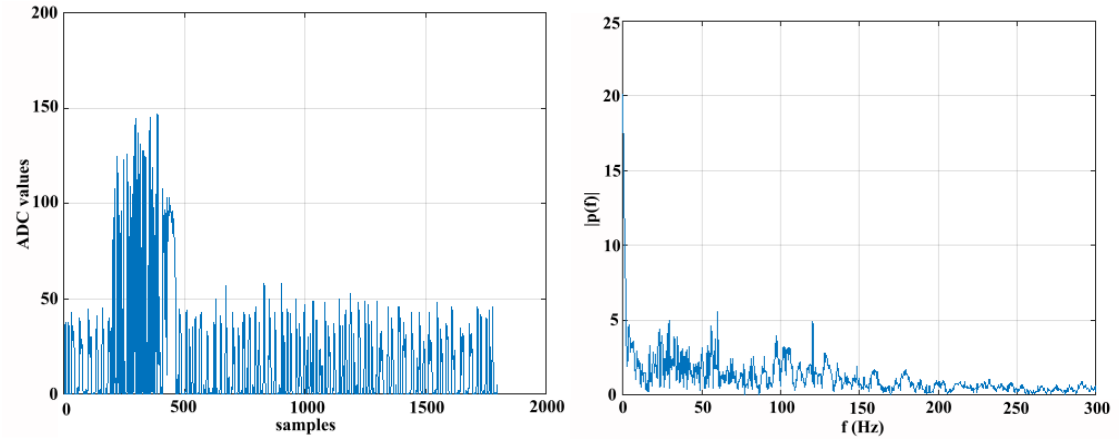

Figure 8-1: Raw EMG Data (Left) and FFT (Right) of Channel 1 Data for “AND”

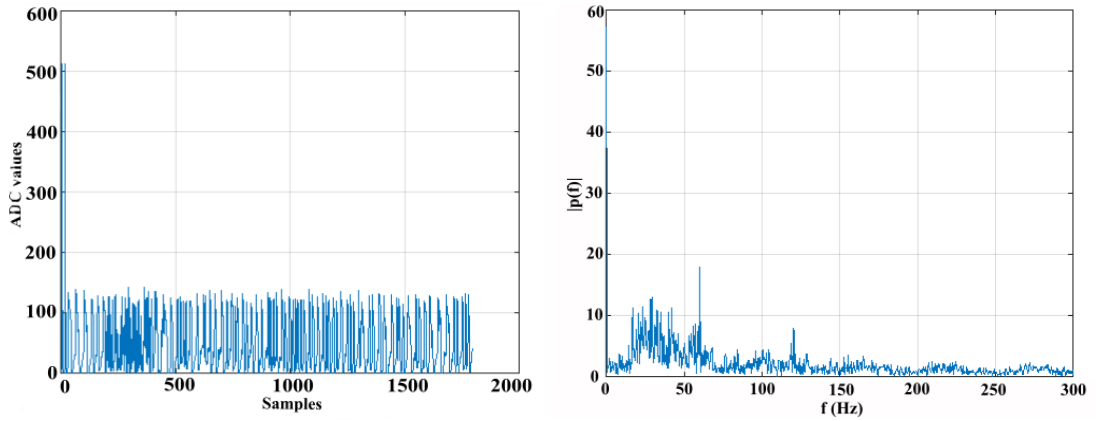

Figure 8-2: Raw EMG Data (Left) and FFT (Right) of Channel 2 Data for “AND”

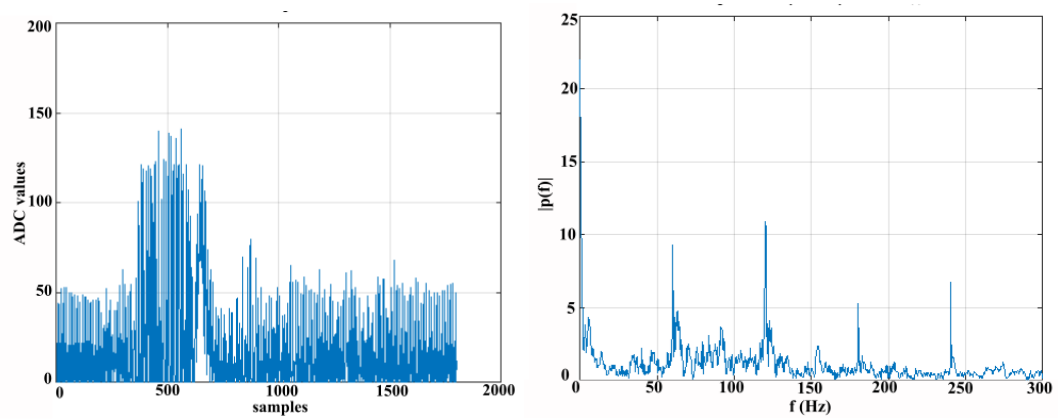

Figure 8-3: Raw EMG Data (Left) and FFT (Right) of Channel 1 Data for “THAT”

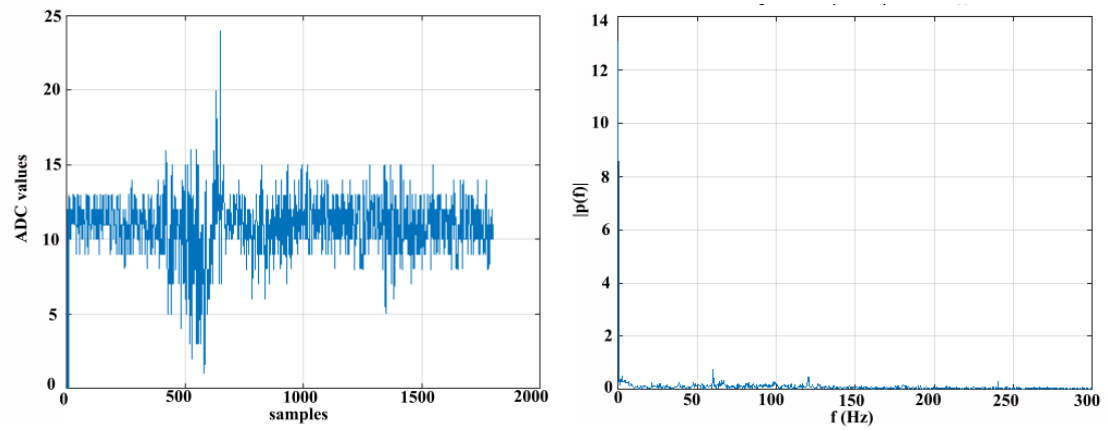

Figure 8-4: Raw EMG Data (Left) and FFT (Right) of Channel 2 Data for “THAT”

The prominent spikes seen on the above graphs signify the successful extraction of the facial EMG signals by the designed hardware. The regions of the highest amplitude represent the utterance of a word. The spikes are dominant within the frequency range of

1 - 100 Hz which verifies the performance of the designed filters within the calculated specifications.

The channel 1 located at cheek region shows conspicuous EMG signals than channel 2 while uttering both words “AND” and “THAT”. Both of the plots have differentiable graphs yet other features are to be realized to distinguish the uttered words. Also, the line noise was encountered at 60 Hz and its harmonics which can be clearly seen in the figure 8-6. This does affect the data but can be avoided using a notch filter with higher order low pass filter. Since the circuit needs to be as compact as possible and additional components will only increase the size of the circuit board, digital filters should be implemented. A digital filter composed of a low pass filter with cut-off at 100 Hz and notch filter at 60 Hz was designed and the output of which is as shown in figure below:

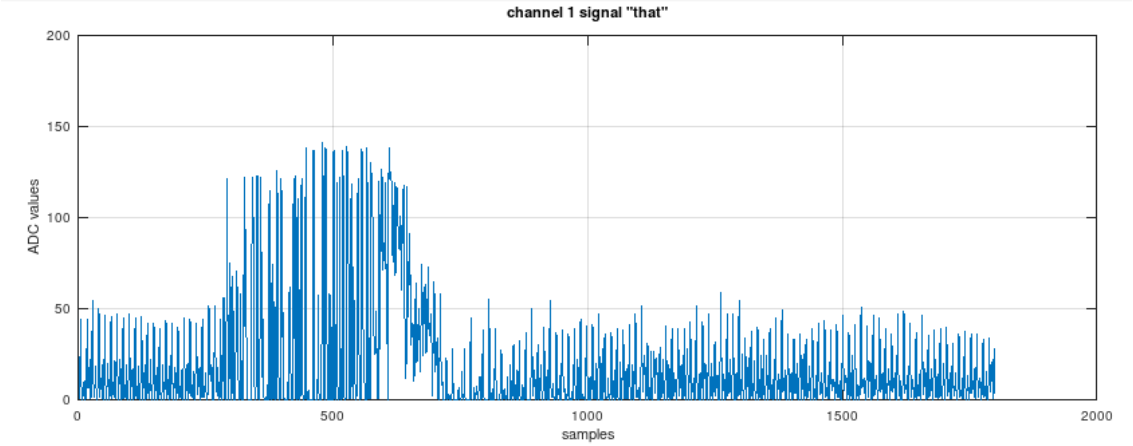

Figure 8-5: Raw Channel 1 EMG Signal of Word “THAT”

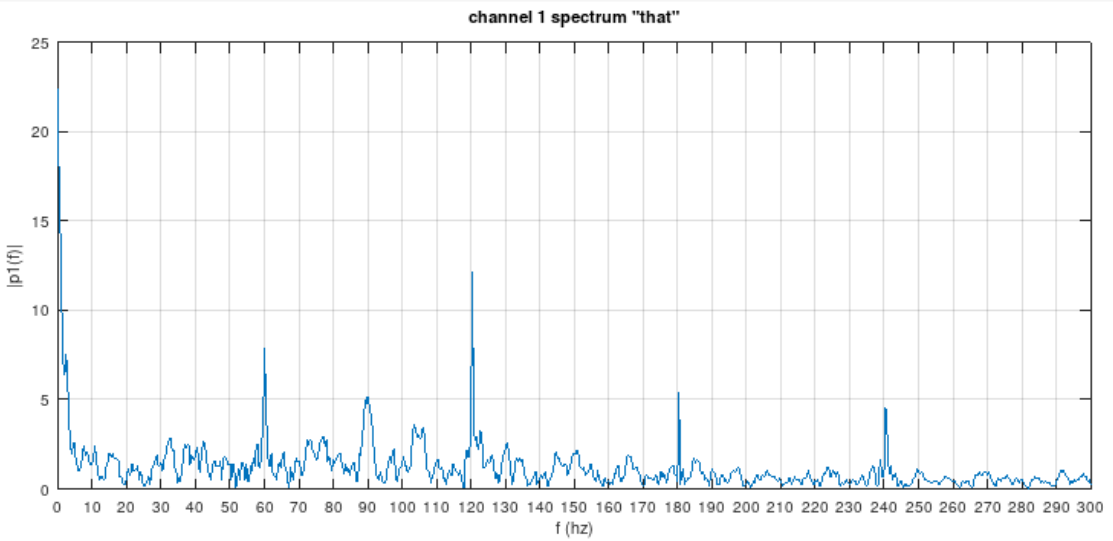

Figure 8-6: Frequency Spectrum of Word “THAT” with Line Noise

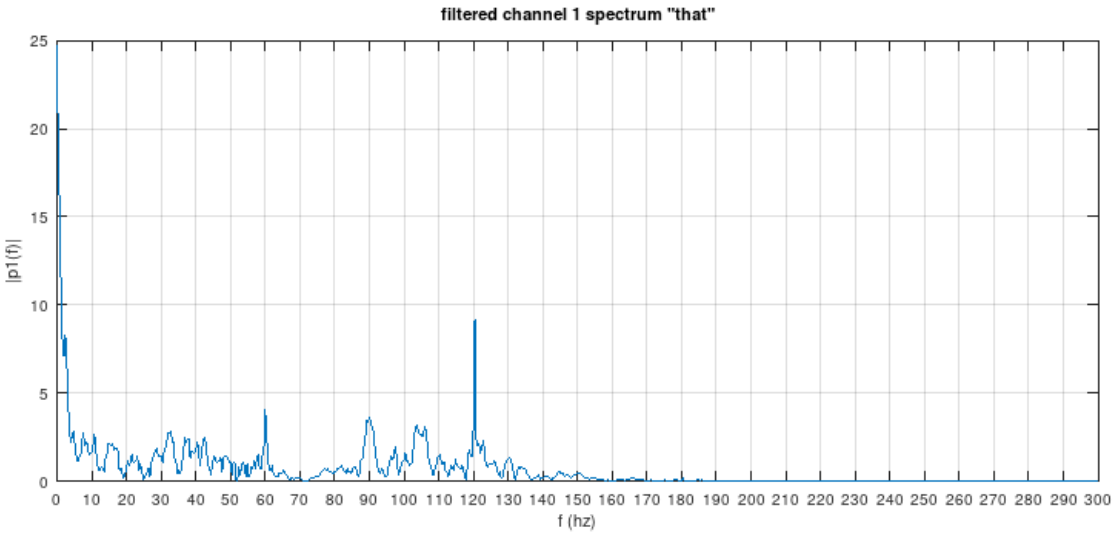

Figure 8-7: Filtered Frequency Spectrum of Word “THAT”

8.1.2 OpenBCI Cyton Board Response

The raw EMG signal recorded using the Cyton Board for the word “call” is shown in the figure below. On visualizing the FFT plot, signals between 1-100 Hz were observed superimposed with line noises and its harmonics at 50 Hz and 100 Hz.

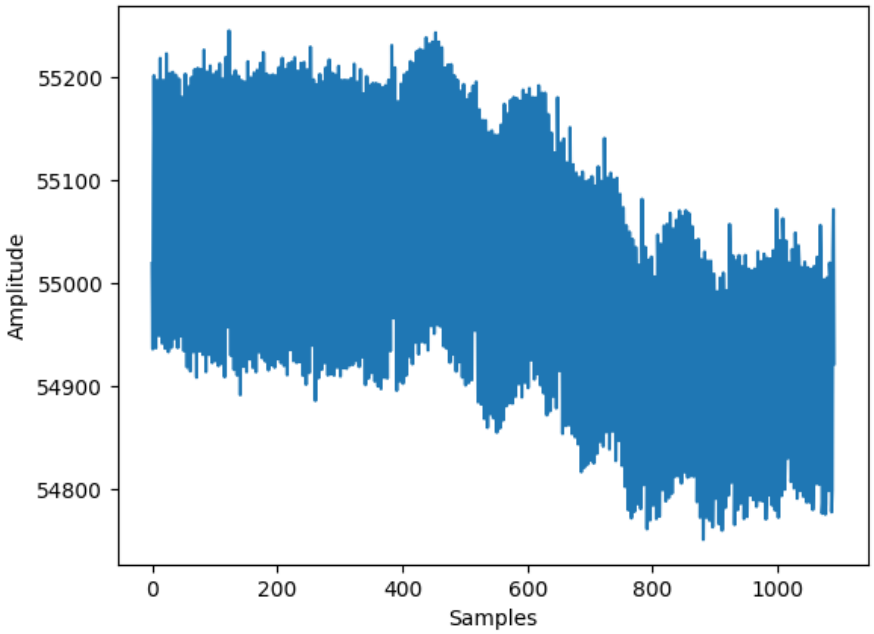

Figure 8-8: Time Domain Signal of Word ‘Call’

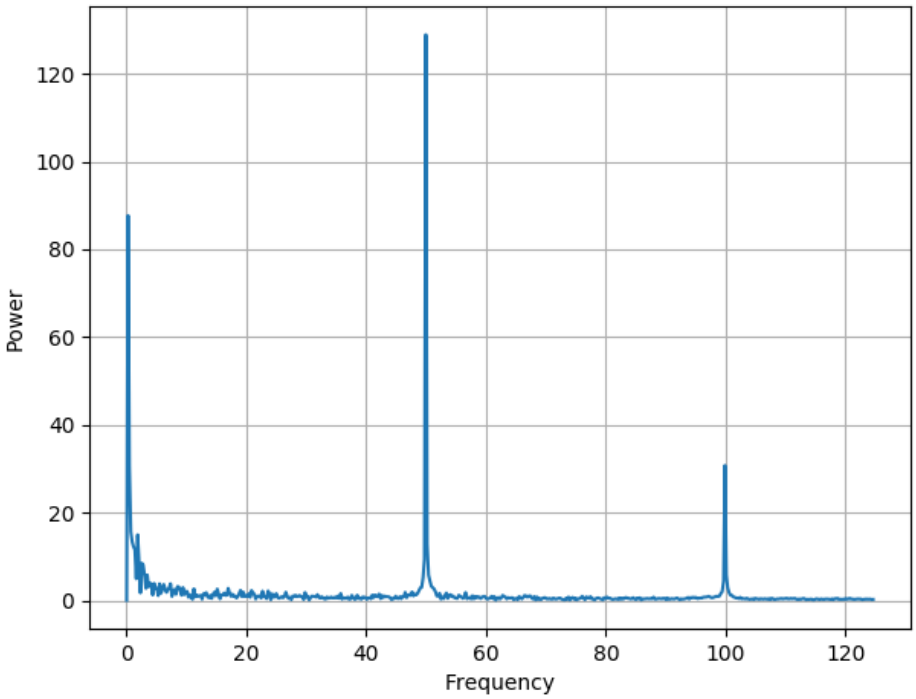

Figure 8-9: Frequency Domain Signal of Word ‘Call’

8.2 Model Response

Initially, the machine learning models were trained using the labelled data parsed from the EMG-UKA corpus and later using self-recorded dataset. The data was prepared according to the data preparation process mentioned in the software implementation section of implementation details. The models were tested using both temporal and spectral features extracted from the data. The temporal features extracted were: Zero Crossing Rate, High Frequency Signals, Rectified High Frequency Signals, Frame Based Power and Double Nine Point Average and the spectral features extracted was STFT. The models selected for the classification tasks were MLP and 1D CNN.

8.2.1. Using EMG-UKA Trial Dataset

The EMG-UKA Trial dataset was primarily analyzed for testing the designed neural networks and tuning the network parameters. This portion is concerned with the analysis of the different models using accuracy, loss graphs and confusion matrices while implementing both the temporal and spectral features.

8.2.1.1 MLP Model Output

The first classifier tested was MLP model with an input layer of 200 nodes, two hidden layers with 64 nodes each and an output layer of 10 nodes that represented the 10 words used in the dataset. The model implemented ReLU as activation function and was trained over 200 epochs with a batch size of 50 using Adam optimizer function with the default learning rate of 0.001. The accuracy for each mode was plotted differently for different speaking modes as shown in the figure below.

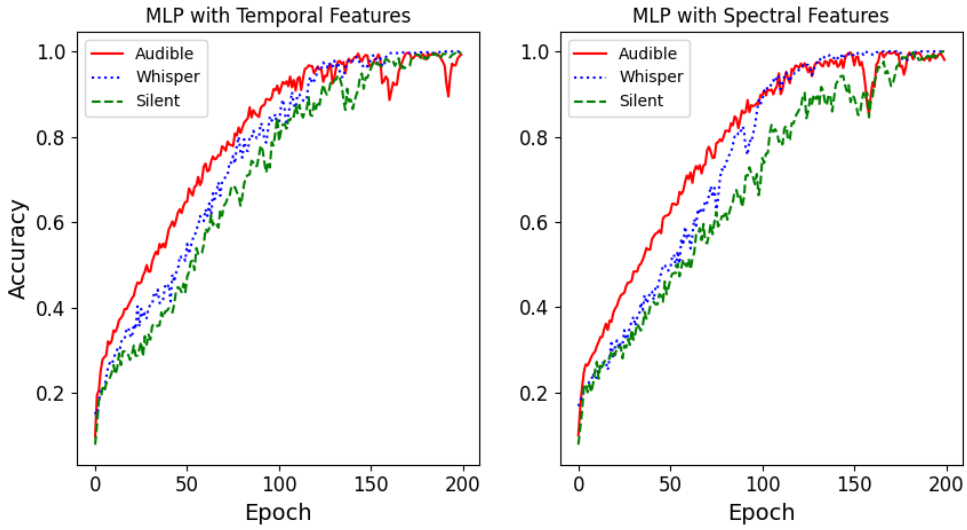

Figure 8-10: MLP Accuracy Curve

From the above figure, it can be deduced that the training accuracy is relatively high for all modes of speaking and for both feature types. The figure on the left is the accuracy plot using the temporal features for three modes represented by; solid line for audible mode, dotted line for whisper mode and dashed line for silent mode. The accuracies for this plot were found to be 99.56%, 98.25% and 97.17% for audible, whisper and silent mode respectively. The figure on the right is the accuracy plot using the spectral features with similar notations used in the figure on the left. The MLP model has the accuracies of 98.99%, 99.12% and 99.37% for audible, whisper and silent mode respectively. From the above figure, it seems that the model is overfitting the data for both spectral and temporal features.

In the figure below, the MLP model output has been studied using the heat map generated from the confusion matrix for silent mode. The data instances for the word “A” and “ARE” have been misclassified mostly as “IN” and many of the word “A” has also been misclassified as “THE” in the model with temporal features. The model implementing spectral features has performed better with respect to the one with temporal features

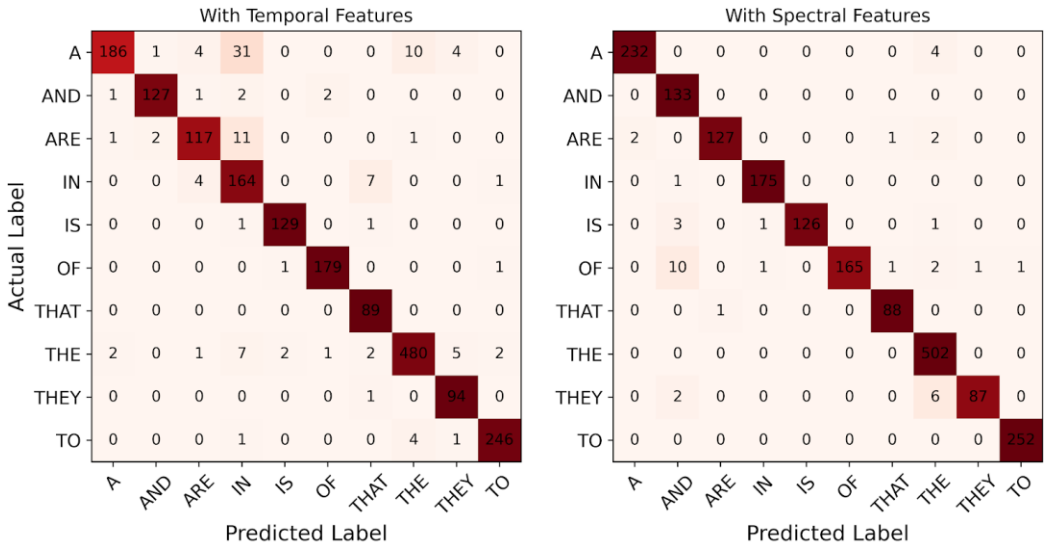

Figure 8-11: MLP Confusion Matrix for Silent Mode

8.2.1.2 CNN Model Output

Later the dataset features were tested using 1D CNN classifier model. The designed model has an input convolutional layer with 100 filters of size 1x3 that is followed by a max pooling layer with a pool size of 1x2. This convolutional layer and max-pooling layer is repeated once which is then followed by a fully connected layer with 100 nodes that is further connected to another fully connected layer with 10 nodes. The activation function that this network utilizes at the convolutional and hidden layers is ReLU and at the output layer is softmax. The model is optimized using Adam optimizer and is trained over 200 epochs with a batch size of 50 with the default learning rate of 0.001. The accuracy for this model is shown in the figure below.

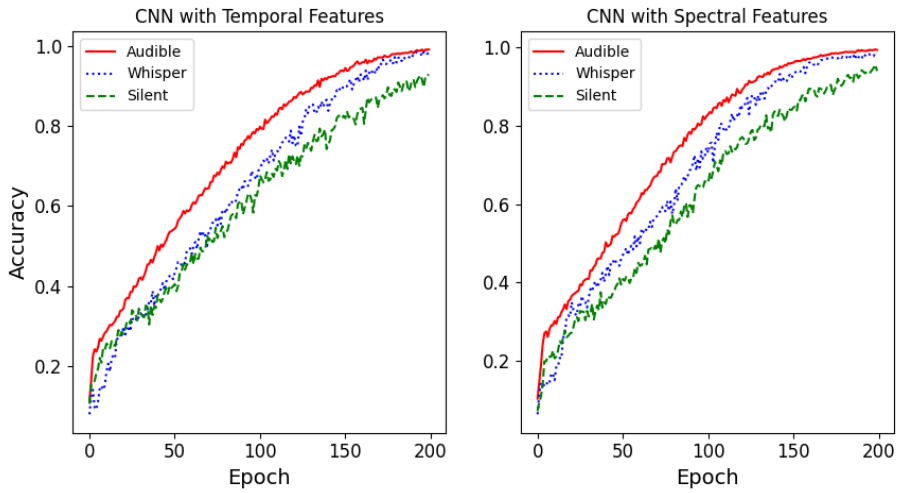

Figure 8-12: CNN Accuracy Curve

In the above figure, the training accuracies for all speaking modes with both temporal and spectral features used for training is shown. As seen similarly in the MLP model, the accuracies are high which leads us to the suspicion that the model is overfitting the data. The accuracies observed are 98.39%, 96.36% and 91.07% for the audible, whisper and silent mode respectively while using temporal features and 99.74%, 99.01% and 92.46% for the audible, whisper and silent mode respectively while using the spectral features.

The CNN model for silent mode was analyzed using a confusion matrix as shown in figure below. From the generated heat maps, it can be inferred that the model using temporal features performed the best.

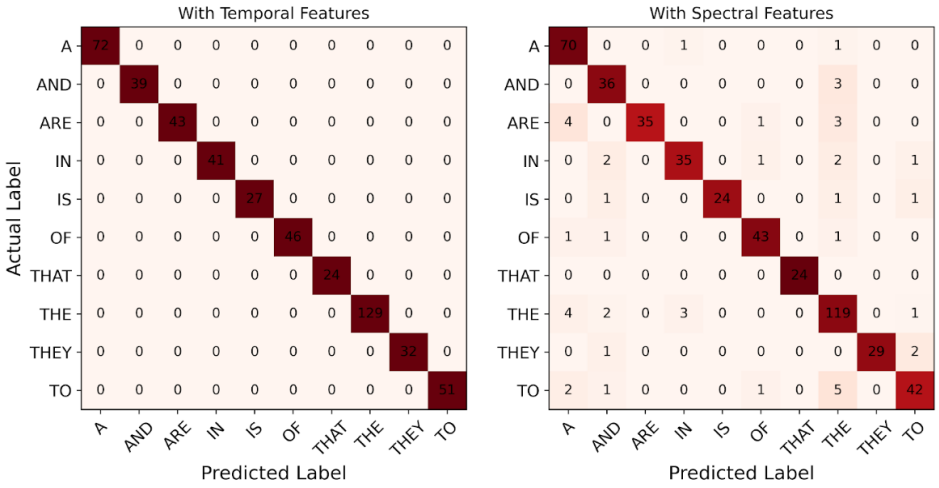

Figure 8-13: CNN Confusion Matrix for Silent Mode

8.2.2 Using Self-Recorded Dataset

After being accessible the OpenBCI hardware, dataset of selected phonetically different words was recorded as described in the section 5.3.2. The temporal as well as spectral features extracted from the dataset were then implemented on the previously designed neural networks with few required modifications.

8.2.2.1 MLP Model Output

MLP model was implemented with an input layer of 200 nodes, two hidden layers with 64 nodes each and an output layer of 10 nodes that represented the selected 10 words used in the dataset. The model implements ReLU as activation function and is trained over 30 epochs with a batch size of 50 using Adam optimizer function with the default learning rate of 0.001. The accuracy and loss for the temporal and spectral features were plotted independently for 'Mentally Rehearsed' mode which is as shown in the figure below.

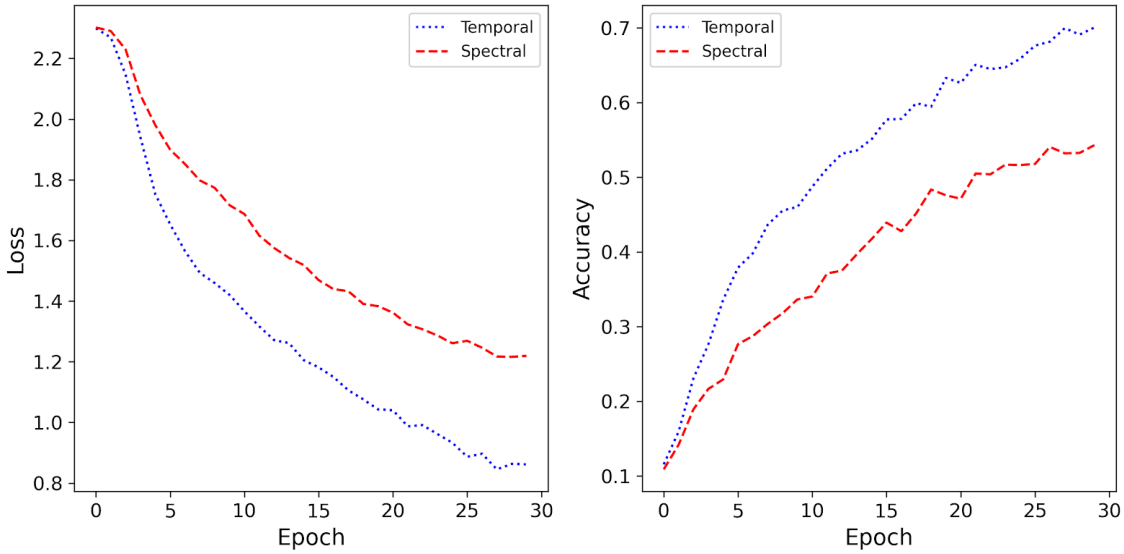

Figure 8-14: Loss/Accuracy vs Epoch Curve (Mentally Rehearsed)

Table 8-1: Accuracy/Loss Comparison of Features in ‘Mentally Rehearsed’ Mode

Features	Accuracy		Loss	
	Train	Validation	Train	Validation
Temporal	0.6892	0.6795	0.8391	1.0856
Spectral	0.6068	0.5550	1.1497	1.4866

The above table illustrates that for ‘Mentally Rehearsed’ mode both train and validation accuracies of temporal features are higher than those of spectral features. Similarly, the train and validation loss of temporal features are lower than those of spectral features. This ultimately implies that the performance of the MLP model with input as temporal features was higher than that with spectral features.

The confusion matrix for the above setting was plotted which is as shown in the figure below. The confusion matrices show that the model is able to predict the words “add”, “subtract” and “you” more accurately than others while using temporal features whereas it predicts the words “stop”, “subtract” and “you” well while using the spectral features.

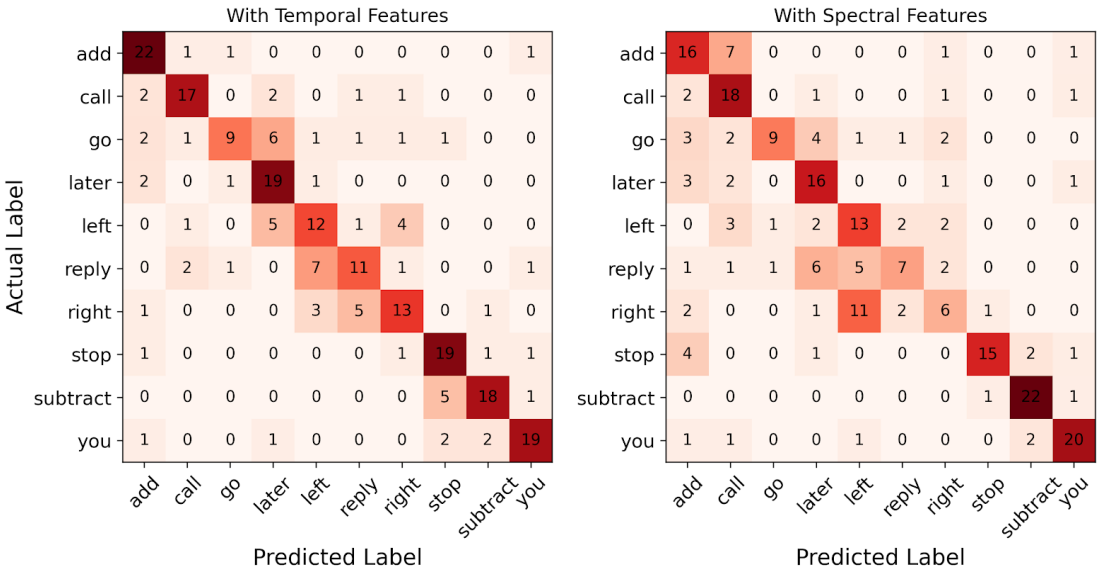

Figure 8-15: Confusion Matrix (Mentally Rehearsed)

The model with spectral features is able to classify the word “stop” more accurately than all other words with a precision score of 88% implying that 88% of the model classification for the word “stop” is accurate. The model seems to be more sensitive towards the prediction of word “subtract” as it has the highest recall score of 92%.

The accuracy and loss using both temporal and spectral features were later plotted differently for ‘Muscle Movement’ mode, the plot of which is as shown in the figure below.

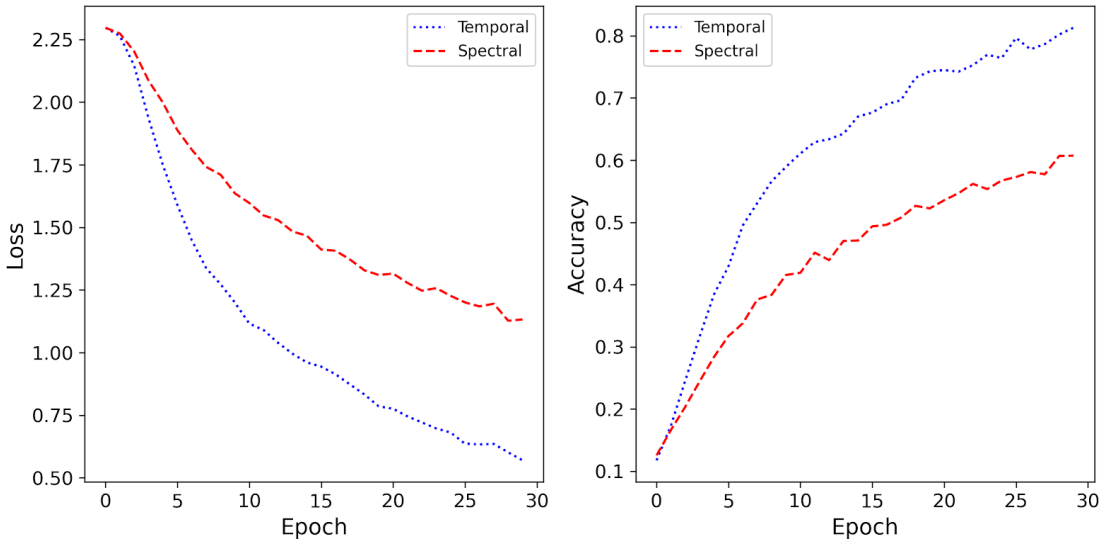

Figure 8-16: Loss/Accuracy vs. Epoch Curve (Muscle Movement)

Table 8-2: Accuracy/Loss Comparison of Features in ‘Muscle Movement’ Mode

Features	**Accuracy**		**Loss**	
	Train	**Validation**	**Train**	**Validation**
Temporal	0.8132	0.7534	0.5701	0.9892
Spectral	0.6074	0.4933	1.1324	1.9535

The above table illustrates that for ‘Muscle Movement’ mode both train and validation accuracies of temporal features are higher than those of spectral features. Similarly the train and validation loss of temporal features are lower than those of spectral features.

This ultimately implies that the performance of the MLP model with input as temporal features was higher than that with spectral features.

The confusion matrix for the above setting was also plotted which is as shown in the figure below. It is observable that the MLP model in ‘Muscle Movement’ mode is capable of predicting words ‘add’, ‘later’, ‘reply’ and ‘you’ more accurately than the rest of the words with input as temporal features. While in case of spectral features, the prediction for words ‘later’, ‘reply’ and ‘you’ seems to be more on the mark.

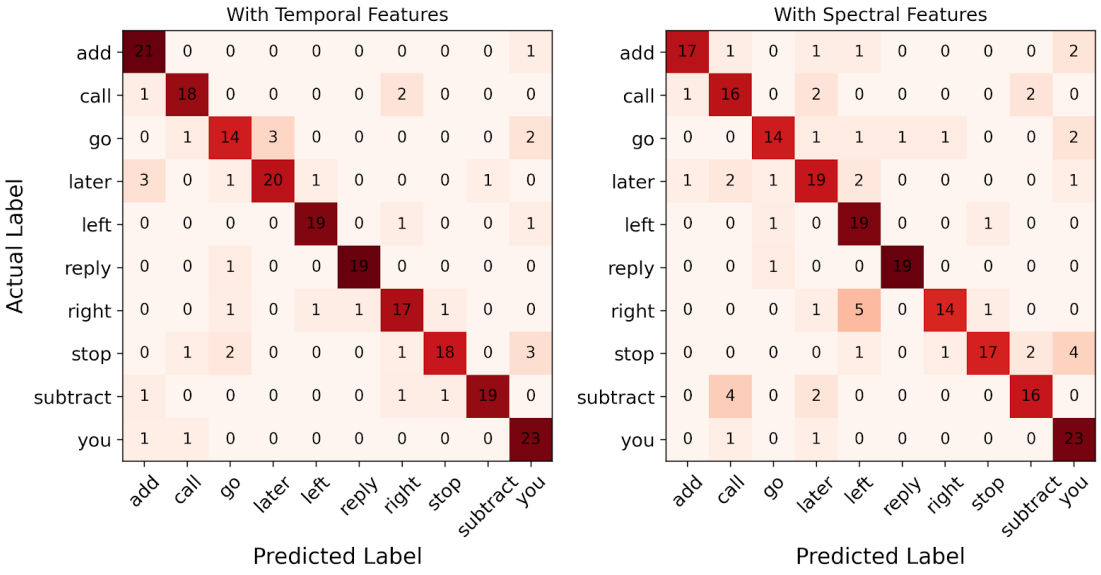

Figure 8-17: Confusion Matrix (Muscle Movement)

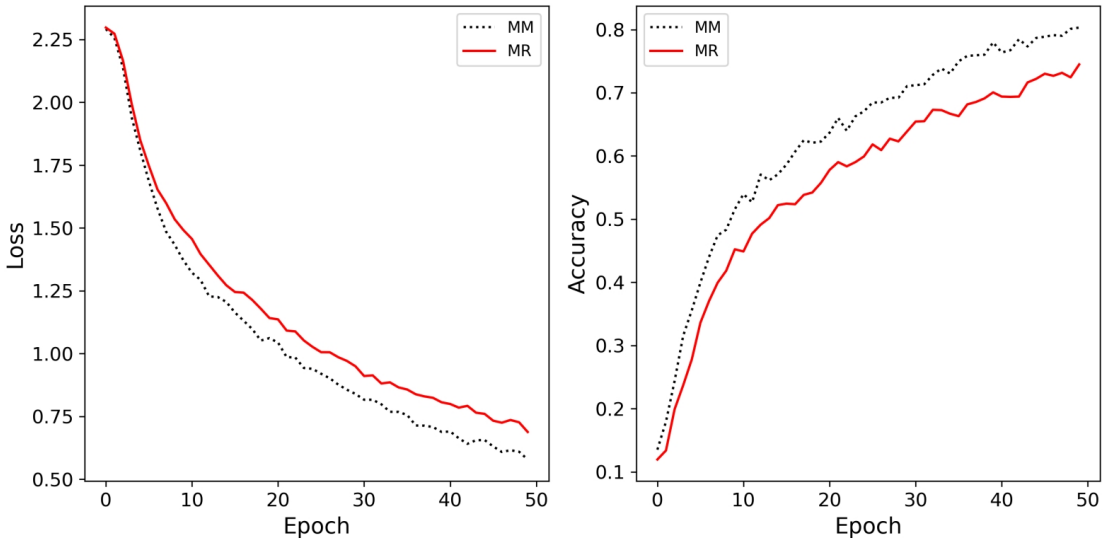

Figure 8-18: Loss/Accuracy vs Epoch (Combined Features)

As shown in the figure above in 'Muscle Movement' mode using both temporal and spectral features combinedly the obtained train loss, train accuracy, validation loss and validation accuracy were as tabulated below:

Table 8-3: Accuracy/Loss Comparison of both Modes for Combined Features

Modes	**Accuracy**		**Loss**	
	Train	**Validation**	**Train**	**Validation**
Muscle Movement	0.8032	0.7265	0.5788	1.5239
Mentally Rehearsed	0.7449	0.6880	0.6883	1.2620

The above table illustrates that for combined features both train and validation accuracies of 'Muscle Movement' mode are higher than those of 'Mentally Rehearsed' mode. Similarly the train and validation loss of 'Muscle Movement' mode are lower than those of 'Mentally Rehearsed' Mode. This ultimately implies that the performance of the MLP model with input as temporal and spectral features combined is higher for 'Muscle Movement' mode.

The confusion matrices for the above configuration were also plotted which are as shown in figures below. For ‘Mentally Rehearsed’ mode combining both temporal and spectral features the confusion matrix showed prominent predictions for words ‘add’, ‘stop’, ‘subtract’ and ‘you’.

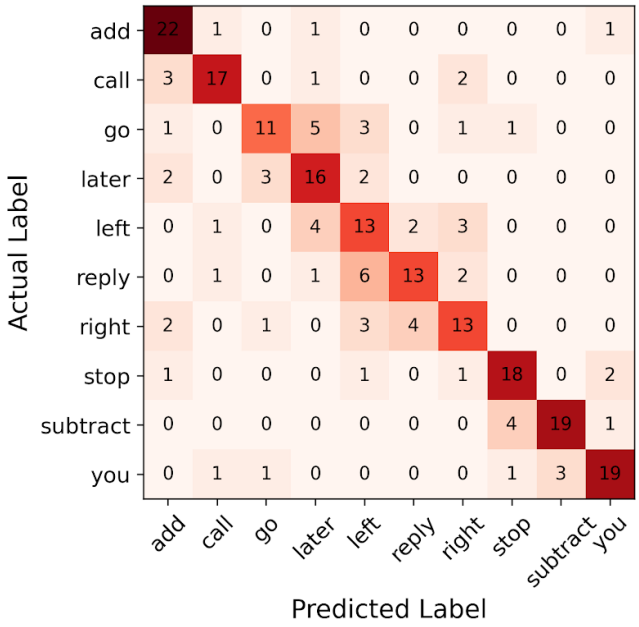

Figure 8-19: Confusion Matrix of Combined Features (Mentally Rehearsed)

While using the combined spectral and temporal features, the highest precision score is for the word “subtract” which gets classified accurately 86% of the time and the highest recall score is 79% for the word “subtract” which implies that out of the total number of the word “subtract”, the classifier detects only 79% of the word “subtract” word by word while 86% are unambiguously classified as “subtract”.

Table 8-4: Precision/Recall with Combined Features (Mentally Rehearsed)

Class	Precision	Recall	F1-Score	Support
add	0.71	0.88	0.79	25
call	0.81	0.74	0.77	23
go	0.69	0.50	0.58	22
later	0.57	0.70	0.63	23
left	0.46	0.57	0.51	23
reply	0.68	0.57	0.62	23
right	0.59	0.57	0.58	23
stop	0.75	0.78	0.77	23
subtract	0.86	0.79	0.83	24
you	0.83	0.76	0.79	25
Macro Avg	**0.70**	**0.68**	**0.69**	**234**
Weighted Avg	**0.70**	**0.69**	**0.69**	**234**

While in case of ‘Muscle Movement’ mode with both temporal and spectral features as input, the confusion matrix was observable with more accurate predictions for words ‘add’, ‘later’, ‘stop’ and ‘you’ than the rest of the utterances.

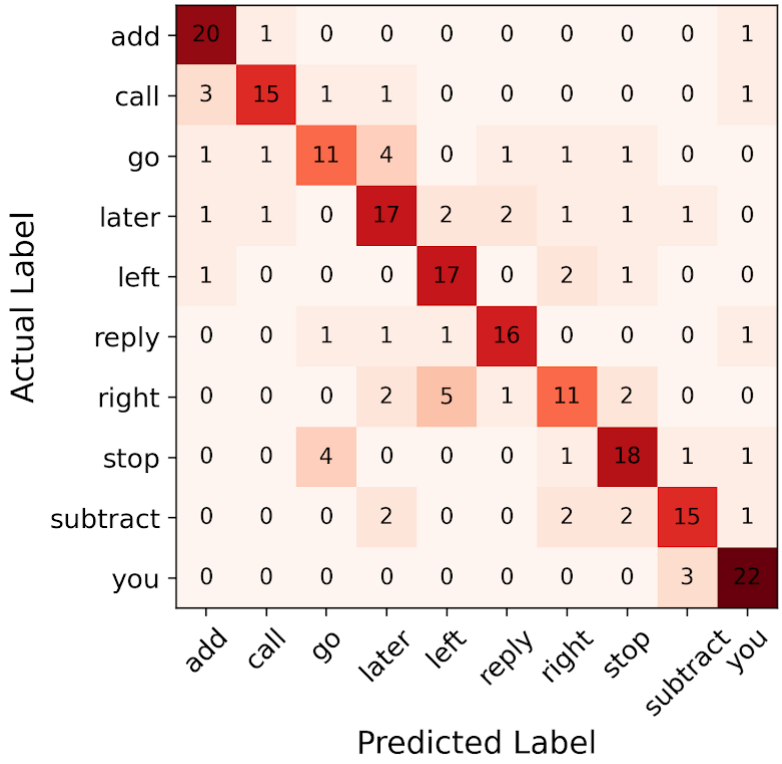

Figure 8-20: Confusion Matrix of Combined Features (Muscle Movement)

While using combinedly the temporal and spectral features, the highest precision score is for the word “you” which gets classified accurately 81% of the time and the highest recall score is 88% for the word “you” which implies that out of the total number of the word “you”, the classifier detects only 88% of the word “you” word by word while 81% are unambiguously classified as ‘you’.

Table 8-5: Precision/Recall with Combined Features (Muscle Movement)

Class	Precision	Recall	F1-Score	Support
add	0.77	0.91	0.83	22
call	0.83	0.71	0.77	21
go	0.65	0.55	0.59	20
later	0.63	0.65	0.64	26
left	0.68	0.81	0.74	21
reply	0.80	0.80	0.80	20
right	0.61	0.52	0.56	21
stop	0.72	0.72	0.72	22
subtract	0.75	0.68	0.71	22
you	0.81	0.88	0.85	25
Macro Avg	**0.73**	**0.72**	**0.72**	**223**
Weighted Avg	**0.73**	**0.73**	**0.72**	**223**

8.2.2.2 CNN Model Output

The final classifier model tested was a 1D CNN. The designed model has an input convolutional layer with 100 filters of size 1x12 that is followed by a max pooling layer with a pool size of 1x2. The convolutional layer with 100 filters of size 1x6 is then implemented which is followed by a max pooling layer with a pool size of 1x2. A fully connected layer with 100 nodes is further connected to another fully connected layer with 10 nodes. The activation function that this network utilizes at the convolutional and hidden layers is ReLU and at the output layer is softmax. The model is optimized using Adam optimizer and is trained over 10 epochs with a batch size of 50 with the default learning rate of 0.001. The accuracy for this model is shown in the figure below.

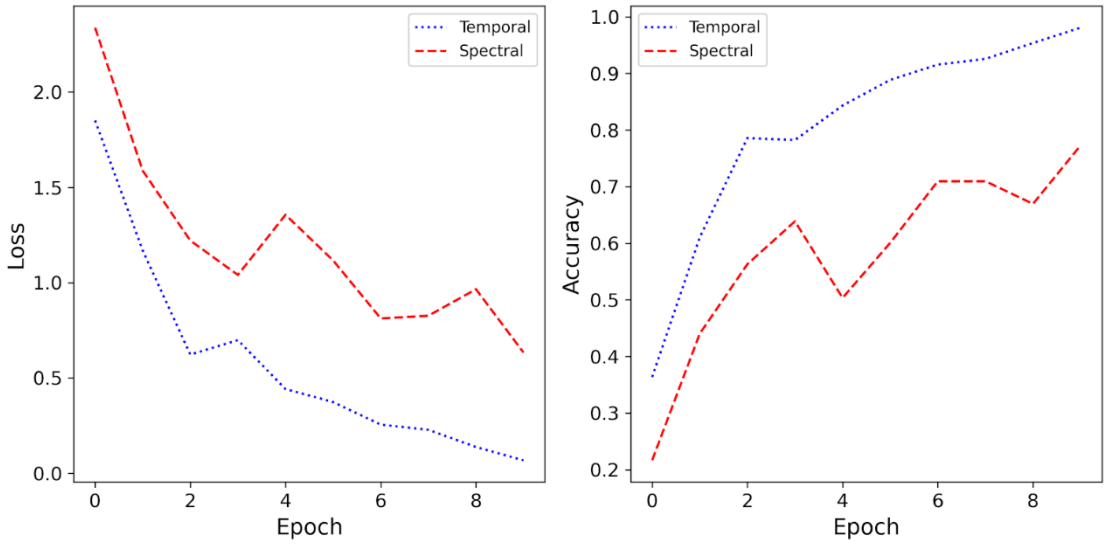

Figure 8-21: Loss/Accuracy vs Epoch Curve (Mentally Rehearsed)

Table 8-6: Accuracy/Loss Comparison of Features in 'Mentally Rehearsed' Mode

Features	**Accuracy**		**Loss**	
	Train	**Validation**	**Train**	**Validation**
Temporal	0.9810	0.8462	0.0677	0.7157
Spectral	0.7730	0.6838	0.6327	0.8935

The above table illustrates that for 'Mentally Rehearsed' mode both train and validation accuracies of temporal features are higher than those of spectral features. Similarly, the train and validation loss of temporal features are lower than those of spectral features. This ultimately implies that the performance of the CNN model with input as temporal features was higher than that with spectral features.

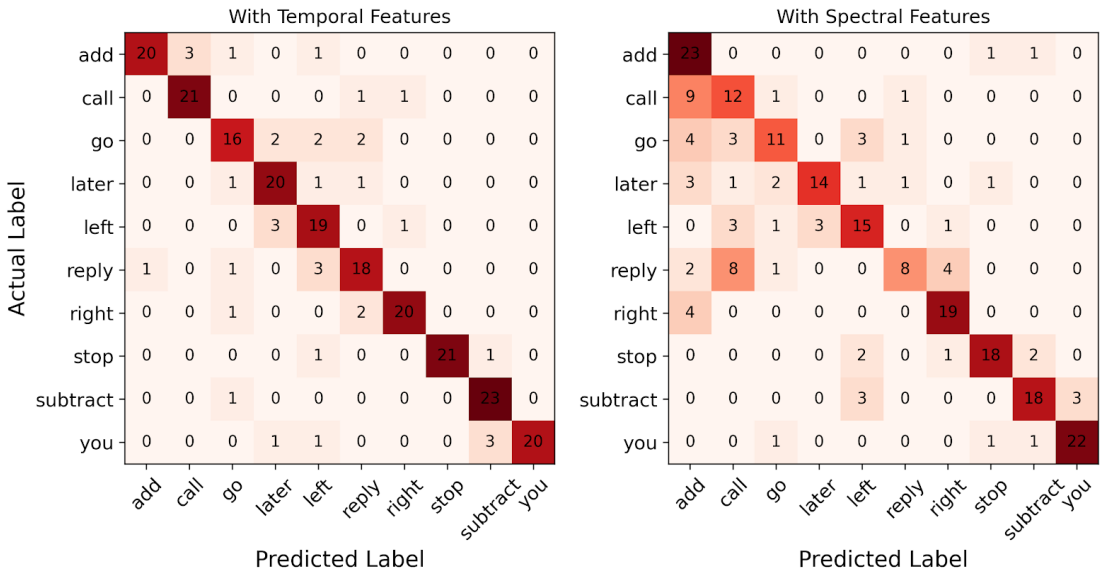

Figure 8-22: Confusion Matrix (Mentally Rehearsed)

The confusion matrices for the above setting were also plotted as shown in the figure above. In ‘Mentally Rehearsed’ mode the confusion matrix plotted for temporal features had more word-for-word predictions for ‘add’, ‘call’, ‘right’, ‘stop’, ‘subtract’ and ‘you’ than those of the remaining words. Similarly, with spectral features under consideration the prediction seems to be promising for ‘add’, ‘right’, ‘stop’, ‘subtract’ and ‘you’. In comparison with the MLP model output under identical configuration the output of CNN model showed more accurate predictions for more utterances trained with less epochs than that of MLP model.

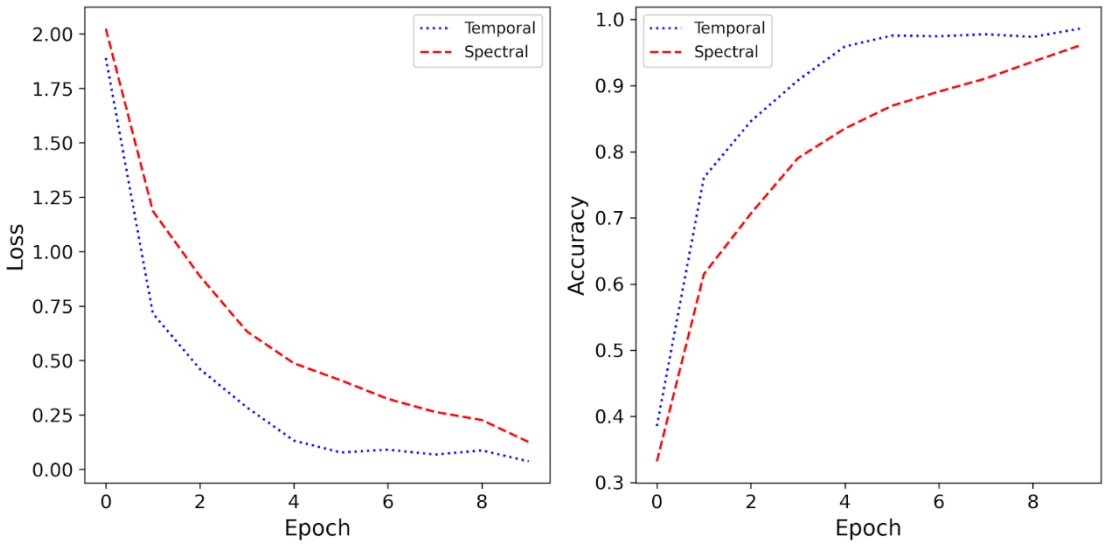

Figure 8-23: Loss/Accuracy vs Epoch Curve (Muscle Movement)

Table 8-7: Accuracy/Loss Comparison of Features in ‘Muscle Movement’ Mode

Features	Accuracy		Loss	
	Train	Validation	Train	Validation
Temporal	0.986	0.843	0.0368	0.7873
Spectral	0.853	0.704	0.4562	1.0572

The above table illustrates that for ‘Muscle Movement’ mode both train and validation accuracies of temporal features are higher than those of spectral features. Similarly the train and validation loss of temporal features are lower than those of spectral features. This ultimately implies that the performance of the CNN model with input as temporal features was higher than that with spectral features.

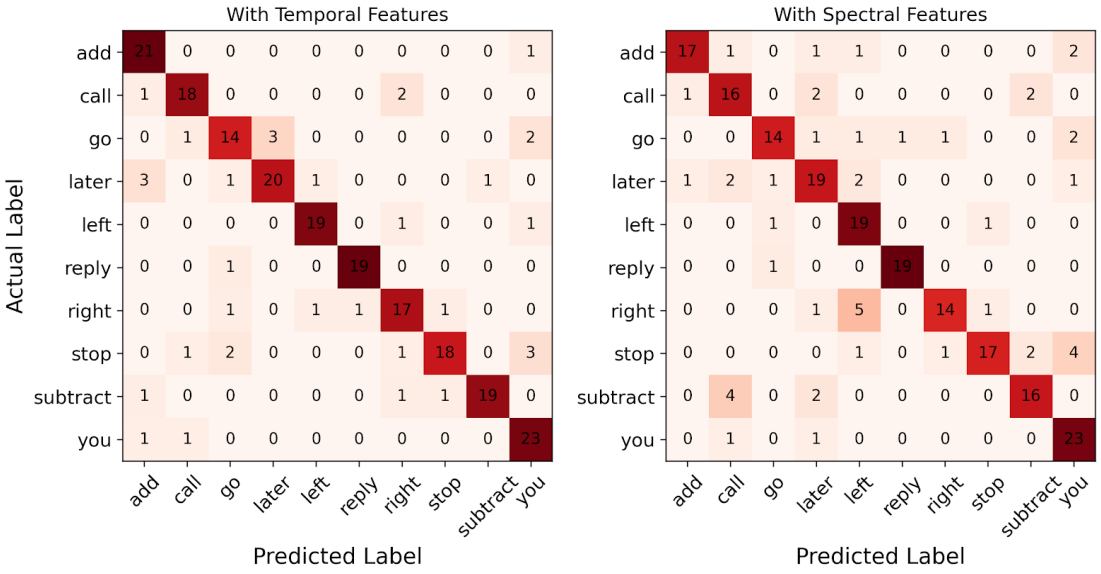

Figure 8-24: Confusion Matrix (Muscle Movement)

The confusion matrices for the above setting were also plotted which are as shown in the figure above. In case of ‘Muscle Movement’ mode the confusion matrix plotted for temporal features had more reliable predictions for ‘add’, ‘later’, ‘reply’, ‘stop’, ‘subtract’ and ‘you’ than those of the remaining words. Similarly with spectral features as input the prediction seems to be more accurate with words ‘later’, ‘left’, ‘reply’ and ‘you’. In

comparison with the output of temporal features as input under identical configuration the output with temporal features showed more prominent predictions for more utterances than that with spectral features in this model.

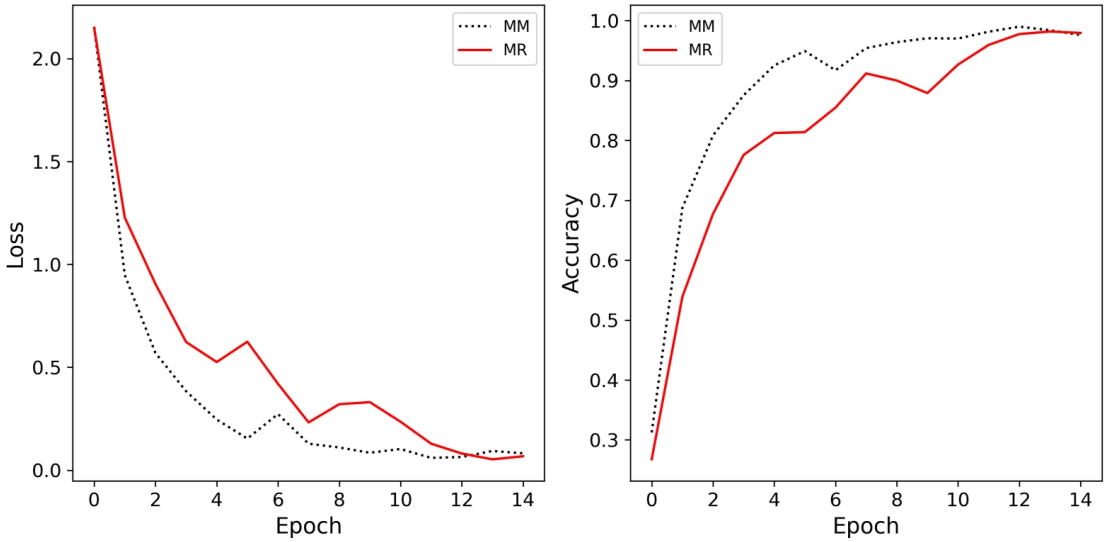

Figure 8-25: Loss/Accuracy vs Epoch Combined Features

As shown in the figure above in 'Muscle Movement' mode using both temporal and spectral features combinedly the obtained train loss, train accuracy, validation loss and validation accuracy were as tabulated below:

Table 8-8: Accuracy/Loss Comparison of Both Modes for Combined Features

Modes	**Accuracy**		**Loss**	
	Train	**Validation**	**Train**	**Validation**
Muscle Movement	0.9761	0.8430	0.0819	1.1073
Mentally Rehearsed	0.9795	0.8333	0.0677	0.8627

The above table illustrates that for combined features both train accuracy of 'Muscle Movement' mode is lower than those of 'Mentally Rehearsedly' mode while for validation accuracy the converse is true. While the train and validation loss of 'Muscle Movement' mode are lower than those of 'Mentally Rehearsedly' mode.

The confusion matrices for the above setting were also plotted which are as shown in figures below. With both temporal and spectral features as input the predictions for the model in ‘Mentally Rehearsed’ mode were more accurate for words ‘add’, ‘call’, ‘subtract’, ‘stop’ and ‘you’ than those of remaining words.

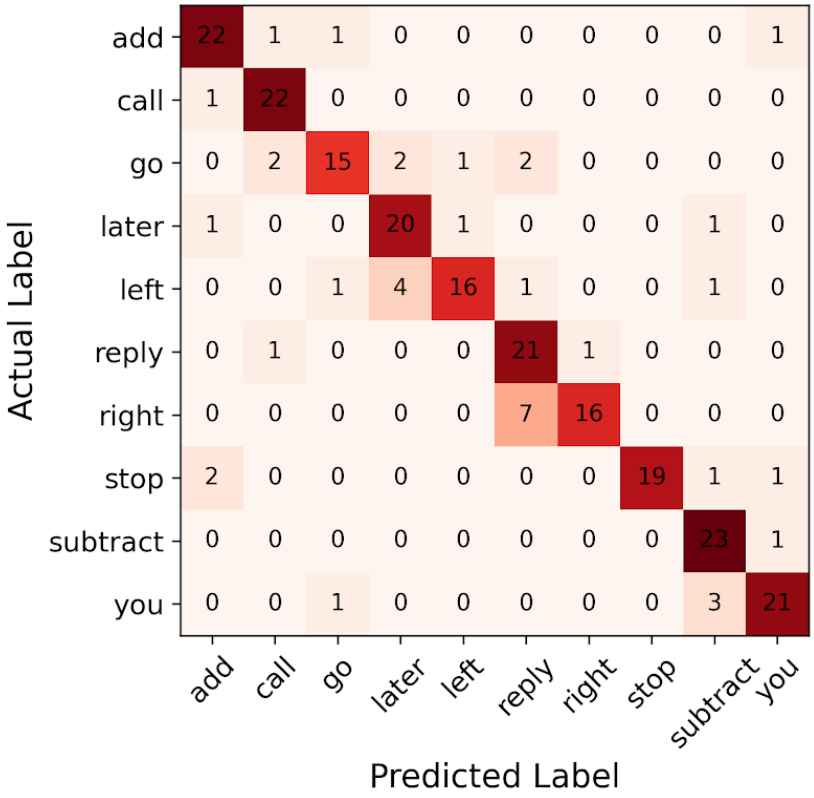

Figure 8-26: Confusion Matrix of Combined Features (Mentally Rehearsed)

The model seems to predict the word “right” with the most accuracy as it only fails to predict the word 6% of the times it predicts given instances as “right”. Similarly, the model has the highest recall score of 96% for the words “call” and “subtract”. It shows that the model is highly biased towards these words and detects them 96% of the time.

Table 8-9: Precision/Recall with Combined Features (Mentally Rehearsed)

Class	Precision	Recall	F1-Score	Support
add	0.85	0.88	0.86	25
call	0.85	0.96	0.90	23
go	0.83	0.68	0.75	22
later	0.77	0.87	0.82	23
left	0.89	0.70	0.78	23
reply	0.68	0.91	0.78	23
right	0.94	0.70	0.80	23
stop	1	0.83	0.86	23
subtract	0.79	0.96	0.87	24
you	0.88	0.84	0.86	25
Macro Avg	**0.85**	**0.83**	**0.83**	**234**
Weighted Avg	**0.85**	**0.83**	**0.83**	**234**

While in the case of ‘Muscle Movement’ mode with same conditions implied the predictions for utterances ‘add’, ‘call’, ‘later’, ‘left’ and ‘you’ were observed to be more unambiguous than the rest of the words.

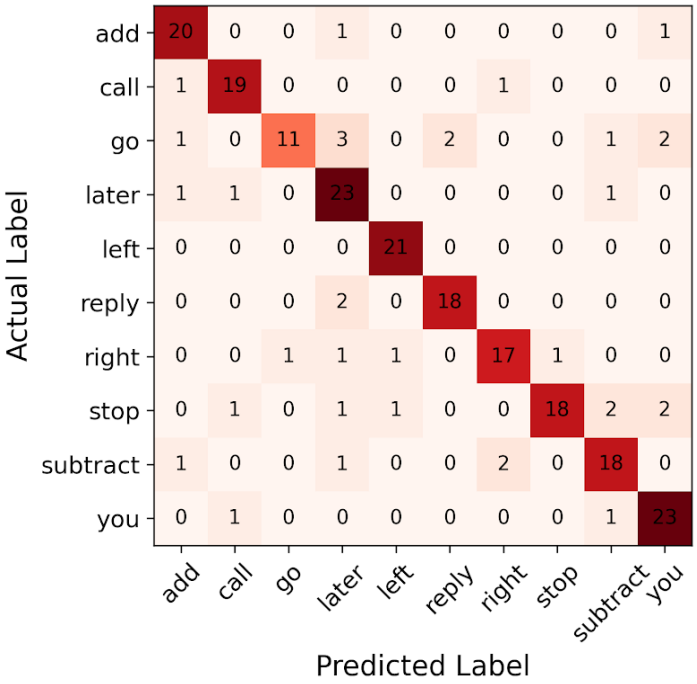

Figure 8-27: Confusion Matrix of Combined Features (Muscle Movement)

The CNN model for “Muscle Movement” mode has the following precision, recall and F1 scores as shown in the table 8-10. The CNN model with combined features is able to classify the word ”stop” more accurately than all other words with a precision score of 95% implying that 95% of the model classification for the word “stop” is accurate and only 5% of the data is misclassified. The model seems to be more sensitive towards the prediction of word “left” as it has the highest recall score of 100%.

Table 8-10: Precision/Recall with Combined Features (Muscle Movement)

Class	Precision	Recall	F1-Score	Support
add	0.83	0.91	0.87	22
call	0.86	0.90	0.88	21
go	0.92	0.55	0.69	20
later	0.72	0.88	0.79	26
left	0.91	1	0.95	21
reply	0.90	0.90	0.90	20
right	0.85	0.81	0.83	21
stop	0.95	0.72	0.82	25
subtract	0.78	0.82	0.80	22
you	0.82	0.92	0.87	25
Macro Avg	**0.85**	**0.84**	**0.84**	**223**
Weighted Avg	**0.85**	**0.84**	**0.84**	**223**

8.3 Model Deployment

After testing the different models on different sets of features, it can be observed that the models perform well when using both types of features and on comparing the results from MLP and CNN model, the performance of CNN model on the dataset seemed more promising. Thus, CNN models trained on both the temporal and spectral features for 15 epochs were saved and deployed on a terminal.

Table 8-11: CNN Model Deployment Result

Speakers	Mode	Samples	Accuracy	Loss
US	Muscle Movement	1765	84.70	0.65
	Mentally Rehearsed	1783	77.29	0.89
SR	Muscle Movement	269	50.56	2.90
	Mentally Rehearsed	276	44.20	2.15
RL	Muscle Movement	235	13.02	8.66
	Mentally Rehearsed	727	9.08	8.19
RN	Muscle Movement	332	9.34	9.55
	Mentally Rehearsed	371	22.37	7.45

For testing purposes, samples of different users were extracted from the dataset and passed to the trained model and the output was visualized on a terminal. Following figure shows terminal output for speakers RL and US in silent and muscle movement mode. Both sample data were tested with both spectral and temporal features.

```
ACCURACY:  0.090784043073654 17
LOSS:  8.198955535888672


Actual Labels   Predicted Lables
   right         right
   stop         right
   call         subtract
   left         call
   later         reply
   left         left
   stop         right
   call         subtract
   you         call
   left         right
   right         reply
   right         left
   right         subtract
   right         left
   stop         you
   left         call
   you         call
   reply         left
   call         add
```

Figure 8-28: Deployed Model on Terminal for RL (Silent Mode)

```
ACCURACY:  0.8470255136489868
LOSS:  0.6574686169624329


Actual Labels   Predicted Lables
   call          left
   go          go
   later          go
   stop          stop
   stop          stop
   later          later
   go          go
   add          add
   left          left
   you          you
   right          right
   stop          stop
   stop          right
   reply          left
   reply          reply
   right          right
   you          you
   later          later
   later          stop
```

Figure 8-29: Deployed Model on Terminal for US (Muscle Movement Mode)

9. ANALYSIS AND DISCUSSION

Analysis of the results obtained from the designed hardware followed by the results from different machine learning models with usage EMG-UKA trial dataset and self-recorded dataset along with the discussions on the phenomena and results are explained in detail within this heading.

9.1 Self-Designed Circuit Analysis

Due to the inherent noises, imprecise values of resistors and capacitors, stray inductance and capacitance, wire resistance and many other parameters, the output of the hardware component was not obtained as ideal expectation rather some magnitude shift, phase shift along with noise were found in the output signal. During the design of amplifier and filter circuits the non-significant values beyond the decimal place were neglected and the resistor value as calculated were unavailable in the market. Moreover, the tolerance of the passive elements like resistor and capacitor were not taken into account during the calculation and design of the circuit. The effect of such neglected parameters were later found while analyzing the circuit in practice. The amplification factor of the amplifiers was attenuated due to which the output signal was attenuated by some factor. Similarly, the roll-off factor of the filters was not as calculated as a result of which transition band was extended and the noises from the frequency bands which were expected to be eliminated by the filter were introduced in the output signal. Also, magnitude as well as phase shifts were found due to imperfections in the filter and amplifier circuit.

The data from the EMG acquisition hardware was sent to laptop through wired connection, which introduced noise in the signals, especially when the charger of the laptop was connected while receiving the data. The crosstalk between the electrode wires also introduced further noise in the system. Furthermore, the Arduino's ADC is unipolar which means it cannot sample the negative voltages from the signal which causes the loss of all the data in the negative domain.

The placement of the signal and ground electrodes has a direct impact on the magnitude and frequency of signal generated. Only a slight displacement in the electrode position

affected the output signal by a large factor. The signal output was also affected by the attenuation introduced in the skin layer that lies between the electrode and the muscle. Also, after prolonged use, the Ag in the Ag-AgCl electrode starts diminishing resulting in weaker ionic potential for the same muscle activity.

9.2 Speech EMG-UKA Dataset Analysis

The dataset for this project has been optimized from the EMG-UKA Trial Corpus. The original dataset was initially collected to perform Automatic Speech Recognition tasks. The corpus has audio signals, corresponding EMG signals and their transcripts in three different modes: Spoken, Whispered and Silent. The EMG signals for each of the modes were trimmed using their transcripts which was not totally accurate. Due to this, the optimized dataset was imperfectly transcribed which introduced major errors in the machine learning models. Another major problem faced was inadequate data in each mode and uneven distribution of data for all the words which ultimately created a bias. This bias made the machine learning models to generalize better for classes with more number of samples and failed to generalize overall. The distribution of data after word-by-word segmentation is shown in the figure 9-1. It can be clearly observed that the number of samples for the words “THE”, “TO” and “A” in all the modes is overwhelmingly high in comparison to other classes.

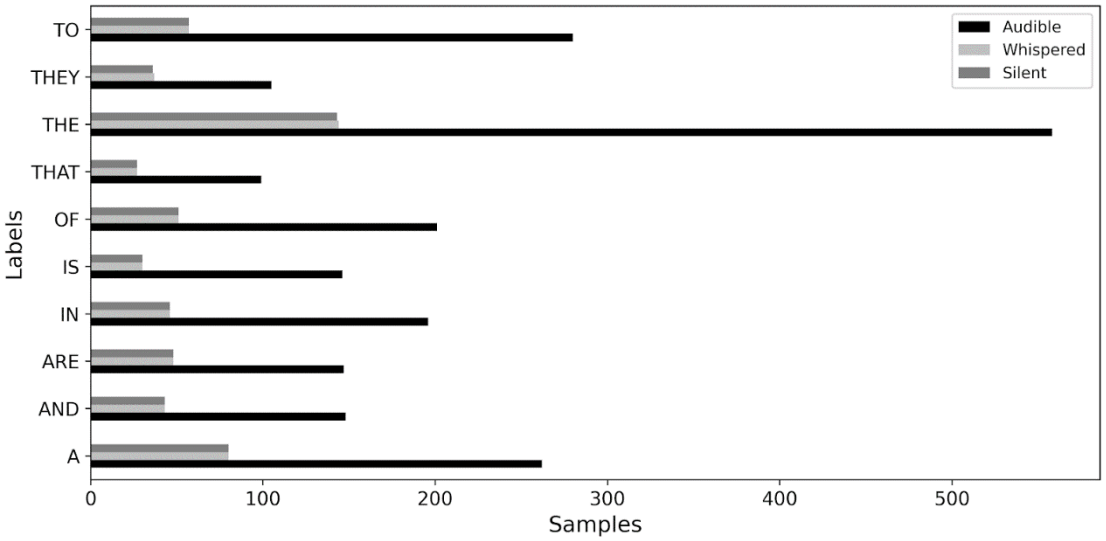

Figure 9-1: Distribution of Data after Segmentation

The summary of all three models for both temporal and spectral features are shown in the tables 9-1, tables 9-2 and table 9-3. The tables contain the train accuracy, test accuracy, precision, recall and F1 score for all variation of the models. From table 9-1, it can be deduced that the best performing model for audible mode is MLP with spectral features as it has the highest precision and recall scores with relatively high test accuracy. However, the MLP model's difference in train and test accuracy is large which makes it essential to analyze the model carefully before considering it as the optimum model. Despite the KNN model with spectral features having the highest test accuracy, the model was not the optimum choice as the model's precision and recall scores are remarkably low.

Table 9-1: Classifier Model Summary Table for Audible Mode

Models	**Feature Type**	**Train Accuracy**	**Test Accuracy**	**Precision**	**Recall**	**F1 Score**
KNN	Temporal	68.76	33.72	0.53	0.48	0.49
MLP	Temporal	99.01	35.81	0.98	0.97	0.97
CNN	Temporal	98.39	39.62	0.99	0.98	0.98
KNN	Spectral	58.01	47.90	0.55	0.52	0.52
MLP	Spectral	100	40.90	1	1	1
CNN	Spectral	99.74	36.28	1	1	1

The table 9-2 shows that the classifying models perform better when using spectral features for the whisper mode. The models with spectral features have marginally high scores in all respects and the optimum model among them is MLP which has the test accuracy of 42.25% and both precision and recall scores as 100%. This model is most likely to be overfitting the data as it has a large difference between the train and the test accuracy.

Table 9-2: Classifier Model Summary Table for Whisper Mode

Models	Feature Type	Train Accuracy	Test Accuracy	Precision	Recall	F1 Score
KNN	Temporal	44.14	26.79	0.44	0.39	0.39
MLP	Temporal	97.88	28.07	0.96	0.99	0.99
CNN	Temporal	96.36	19.28	0.91	0.85	0.87
KNN	Spectral	45.25	31.57	0.40	0.37	0.38
MLP	Spectral	100	42.10	1	1	1
CNN	Spectral	99.01	28.07	0.99	0.99	0.99

From table 9-3, it can be seen that the MLP and CNN models have a similar performance in terms of accuracy. The KNN model has good test accuracies but the precision and recall scores are much lower. The optimum model for the silent mode seems to be MLP model with spectral features as it has good precision and recall scores in comparison to CNN. Further analysis of both CNN and MLP models is required before selecting between the two.

Table 9-3: Classifier Model Summary Table for Silent Mode

Models	Feature Type	Train Accuracy	Test Accuracy	Precision	Recall	F1 Score
KNN	Temporal	41.52	28.11	0.38	0.32	0.31
MLP	Temporal	89.57	21.05	0.94	0.93	0.94
CNN	Temporal	91.07	19.39	0.90	0.89	0.89
KNN	Spectral	43.45	29.82	0.41	0.37	0.37
MLP	Spectral	99.05	24.56	0.99	0.99	0.99
CNN	Spectral	92.46	24.61	0.94	0.92	0.93

9.3 Self-Recorded Dataset Analysis

The self-recorded dataset consists of sub-vocal utterances in two modes; ‘Muscle Movement’ and ‘Mentally Rehearsed’ for each of the speakers. The sample distribution of different speakers in both the speaking modes is as shown in figure below. In figure, MM represents ‘Muscle Movement’ mode and MR represents ‘Mentally Rehearsed’ mode.

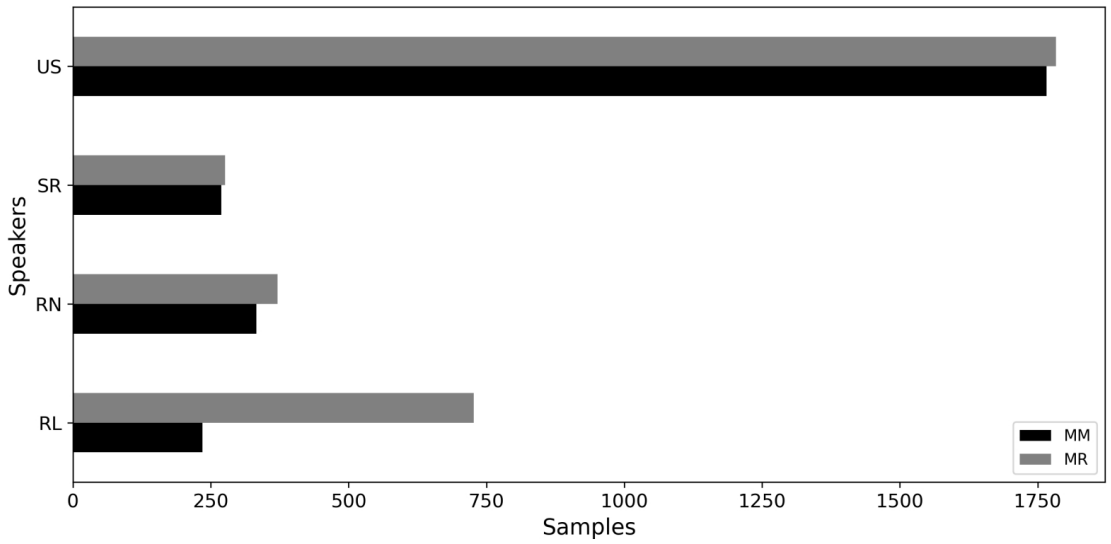

Figure 9-2: Sample Distribution of Speaker

The dataset was recorded considering phonetically different words which in this case are; ‘Add’, ‘Call’, ‘Go’, ‘Later’, ‘Left’, ‘Reply’, ‘Right’, ‘Stop’, ‘Subtract’ and ‘You’. The distribution of these words in both speaking modes is as shown in figure below.

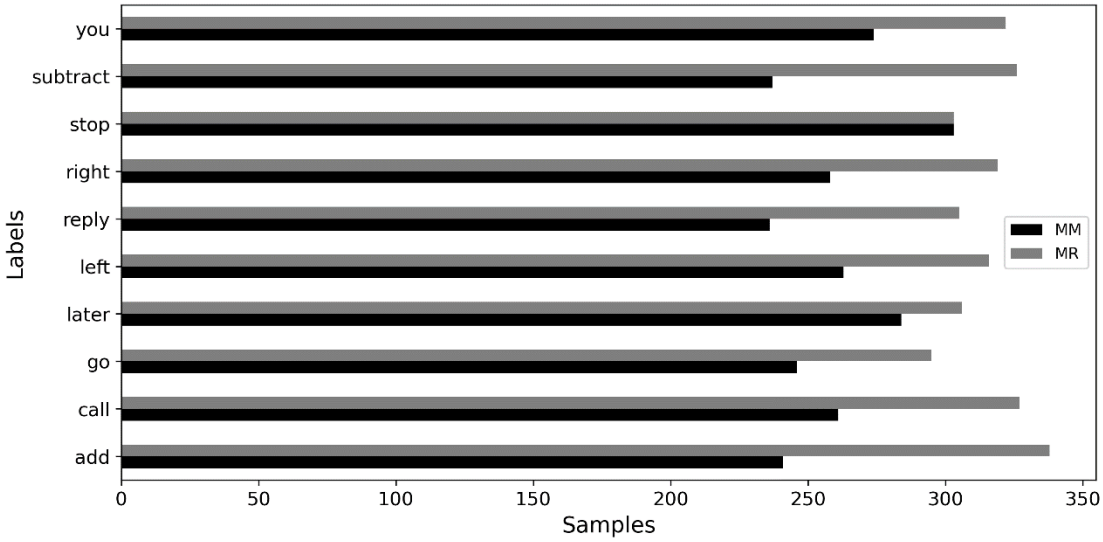

Figure 9-3: Sample Distribution of Labels

The overall summary of the models implemented with various parameters is as tabulated below. As shown in the table the accuracy, precision score, recall score and F1 score are higher in ‘Muscle Movement’ mode than those in ‘Mentally Rehearsed’ mode in both MLP and CNN models. Movement of the muscles where the electrodes were placed was prominent during the recording of signals in ‘Muscle Movement’ mode. In contrast to this, in ‘Mentally Rehearsed’ mode there was no explicit movement of muscles as the words were internally articulated. Thus the abundance of the signal strength in ‘Muscle Movement’ mode added the accuracy of the models.

Comparing the outputs of MLP and CNN models the accuracy, precision score, and recall score along with F1 score was obtained higher for CNN model than those for MLP model. Moreover, the number of epochs the data trained in MLP model was 30 epochs for temporal and spectral features independently and that for both features combined was 50 epochs while in case of CNN model the number of epochs the data trained was 10 epochs for temporal and spectral features independently and that for both features combined was 15 epochs. This huge variation in accuracy and required number of epochs for convincing outputs was due to the fact that MLP model is a fully connected model with dense connection of neurons resulting redundancy and inefficiency whereas in CNN model the neurons are sparsely connected reducing redundancy. Also, the parameters like kernel size, stride size and padding allow the parameter sharing and weight sharing as a result

of which CNN is capable of recognizing a specific pattern anywhere within the window unlike in case of MLP model.

Table 9-4: Model summary

Model	Mode	Feature Type	Accuracy	Validation Accuracy	Precision	Recall	F1 Score
MLP	Muscle Movement	Temporal	81.32	75.34	0.75	0.75	0.75
		Spectral	60.74	49.33	0.49	0.49	0.47
		Temporal and Spectral	80.32	72.65	0.73	0.72	0.72
	Mentally Rehearsed	Temporal	68.92	67.95	0.69	0.68	0.67
		Spectral	60.68	55.50	0.63	0.60	0.60
		Temporal and Spectral	74.49	68.80	0.70	0.68	0.69
CNN	Muscle Movement	Temporal	98.60	84.30	0.85	0.84	0.84
		Spectral	85.30	70.40	0.80	0.78	0.78
		Temporal and Spectral	97.61	84.30	0.85	0.84	0.84
	Mentally Rehearsed	Temporal	98.10	84.62	0.85	0.85	0.85
		Spectral	77.30	68.38	0.71	0.68	0.68
		Temporal and Spectral	97.95	83.33	0.85	0.83	0.83

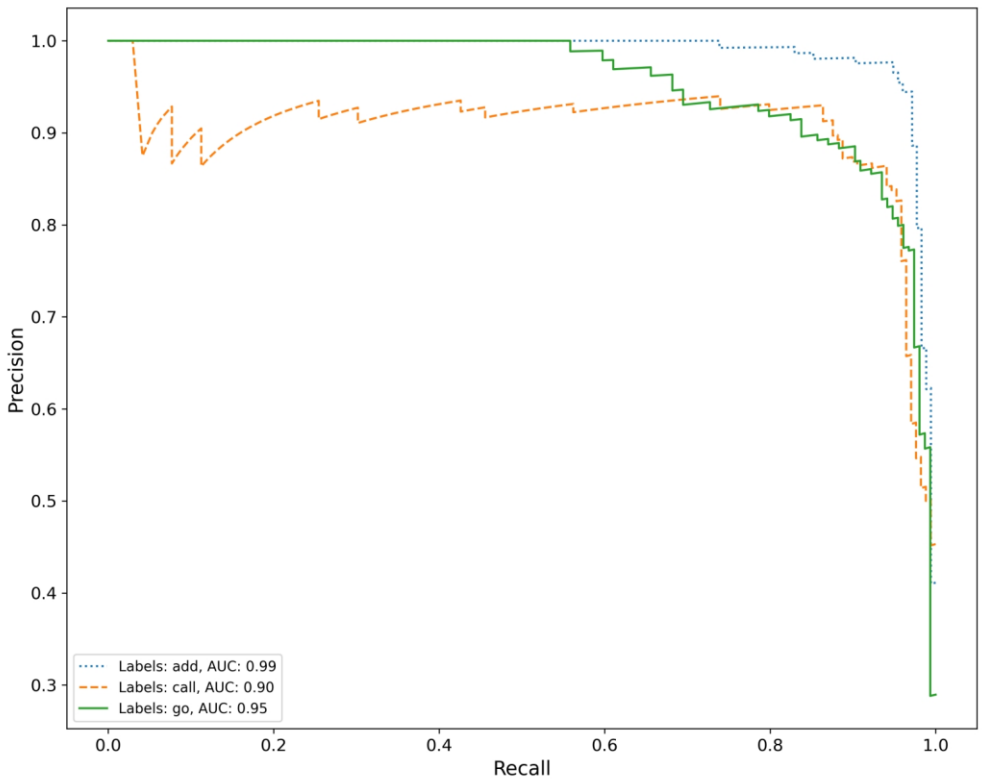

Figure 9-4: Precision Recall AUC Curve (US ‘MM’ Mode)

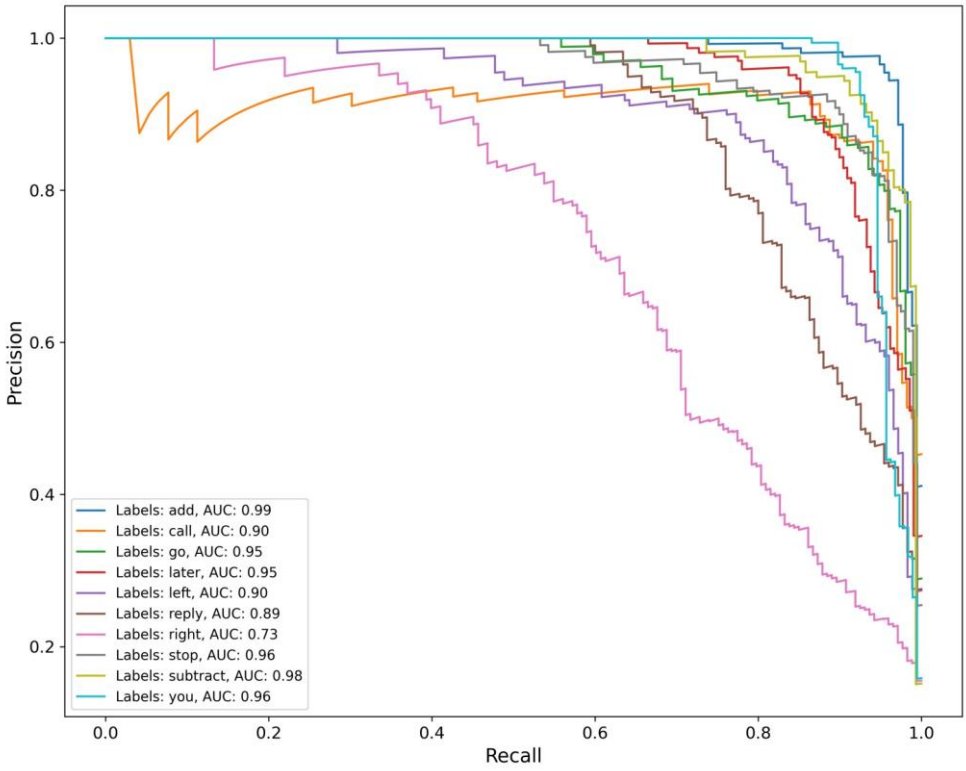

Figure 9-5: Precision Recall AUC Curve

The precision recall curve of all the labels uttered by speaker ‘US’ in MM-mode is as shown in the figure above. The curves for all labels are above the horizontal line at the bottom of the curve and are also above the diagonal line. This implied a skilled-classifier which means that the classifier is able to classify these labels with greater accuracy. As shown in the graph the classifier has the highest AUC value for label ‘Add’ which is 0.99 and the lowest is for ‘You’ which is 0.96. All the values are very dominant which implies the high-accuracy performance of the model.

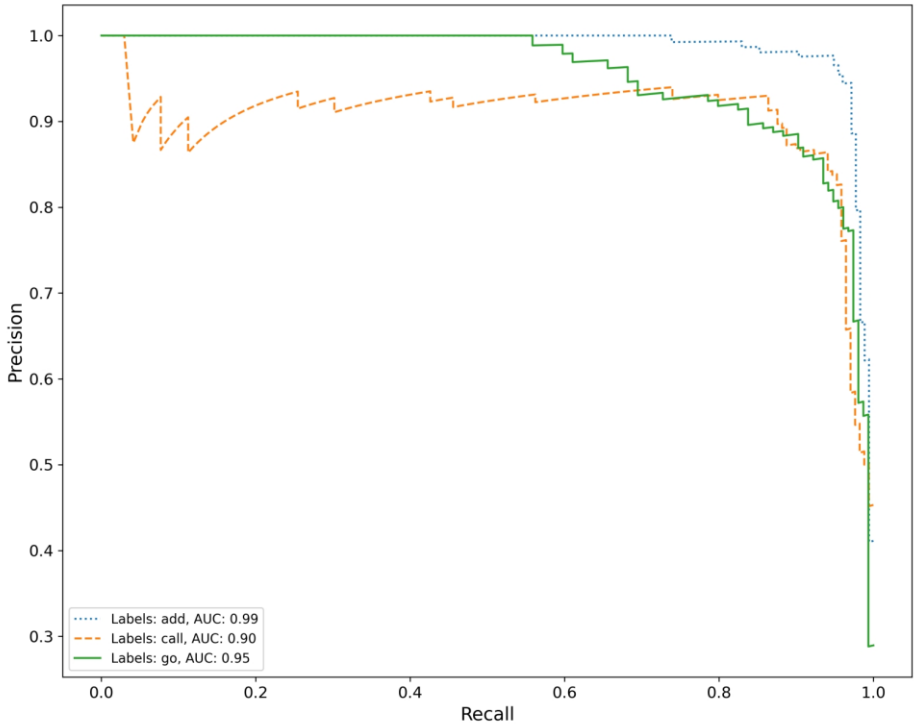

Figure 9-6: Precision Recall AUC Curve (Labels 123)

The ROC curve of the labels ‘Add’, ‘Call’ and ‘Go’ uttered by speaker ‘US’ in MM-mode is emphasized in the figure above.

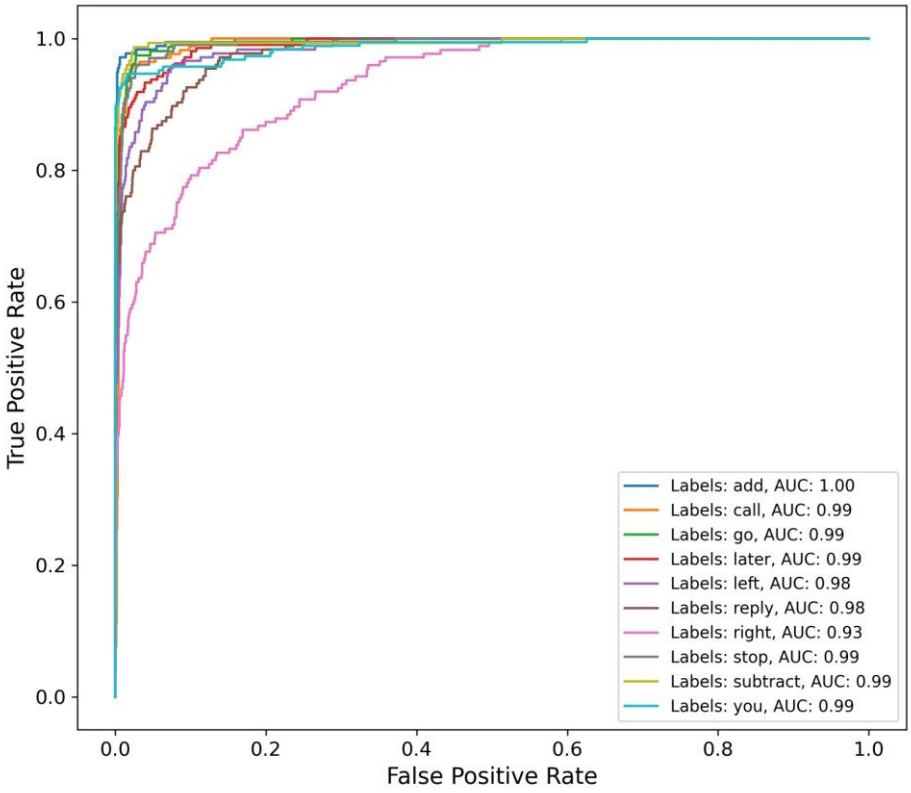

Figure 9-7: ROC AUC Curve

The Receiver Operating Characteristics (ROC) curve for all classes (labels) of speaker 'US' in 'Muscle Movement' mode is as shown in figure above. The curves of all the classes are above the diagonal line. This does show that the labels are classified well. The classification of the word 'right' is less and has AUC value 0.93. On observing the curve the model seems to be predicting the word 'add' much better than other labels and has AUC value 1.00 so the probability for its prediction is greater.

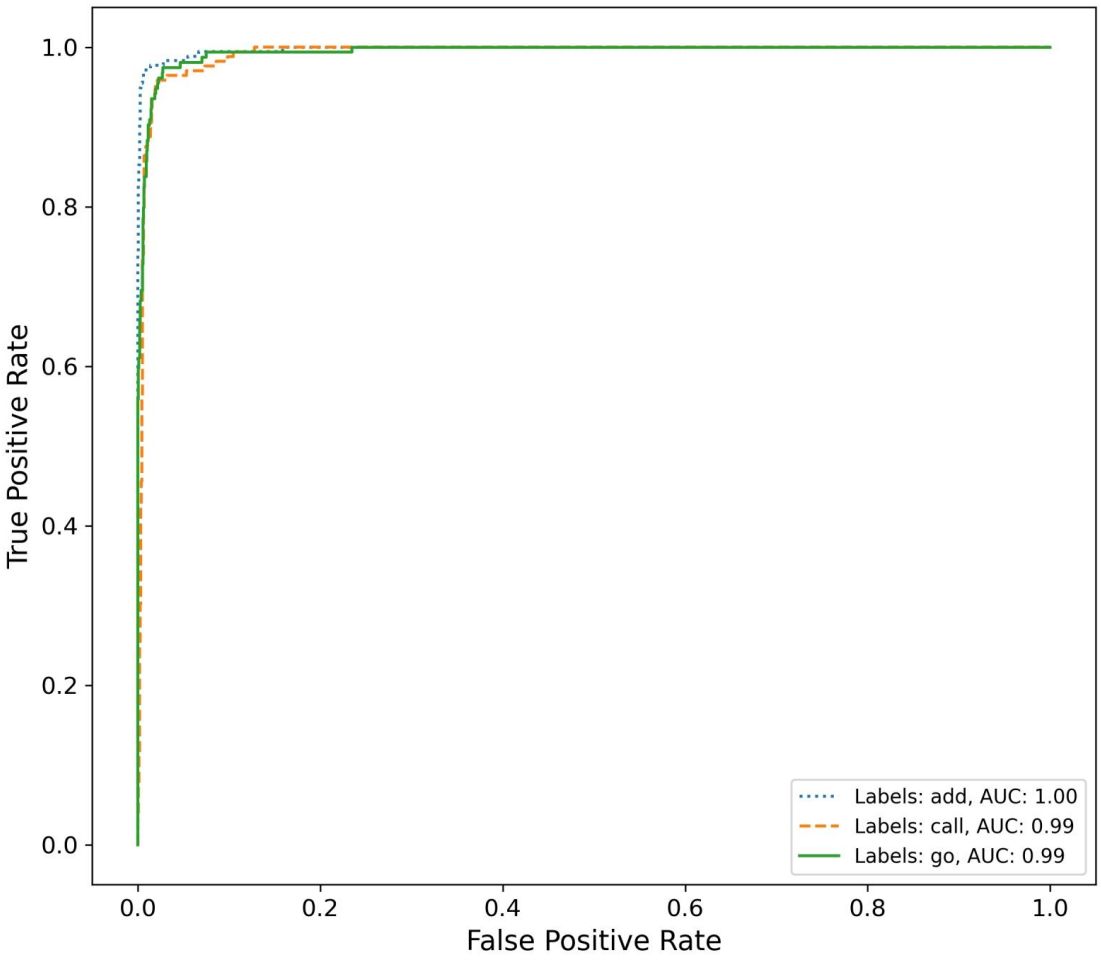

Figure 9-8: ROC AUC Curve (Labels 123)

The models show good accuracy for both training and testing but the distribution of data is imbalanced. Thus the accuracy as shown might not be appropriate and the models might fail to perform as expected. For a better performance measure precision, recall and F1-scores were noted, the model does give decent scores and shows promising results but still the limited numbers of data for different users is a must factor to be considered while analyzing these scores.

The random and user dependent nature of EMG is also a factor to be considered while deploying the model so the accuracy and prediction of users might differ for different users.

10. FUTURE ENHANCEMENT

The self-designed hardware can be improved by designing analog filters with higher roll off and implementing on-board notch filters using components specifically designed for low noise and biosignal. ADC with higher resolution and sampling rate could also aid the accuracy of the system. Even with the use of better hardware i.e. Cyton board, which is compact and portable yet the use of an 8 channel electrode does pose some discomfort to the user thus selecting fewer muscles could make it more convenient for the users.

The collected dataset were recorded solely for silent articulations, which could be further enhanced by recording with audible articulation of words. This enables to distinguish the instance of utterance of the words. It also features alignment of the articulated words and enables exact labelling of the instance of articulation. The length of utterance is different for different users so word rate can be taken into account. One could separate the articulation in phonemes. With this, phoneme based classification could be achieved. Instead of recording discrete words, continuous speech can be recorded. Thus extending the project from discrete word recognition to continuous speech recognition.

The volume of the dataset can be extended by extracting signals from speakers of more diverse age groups belonging to both genders. The signal extraction procedure can also be conducted in various sessions which introduces variation in data making the data session independent. Moreover, it aids in reducing the effects of muscle fatigue. CNN and MLP are better at recognizing complex features of data. The RNN and LSTM which are popular for speech recognition and prediction can also be implemented for silent articulation. Mainly models that perform better for the time based data are more preferred. The project can further be extended by implementing the features of seamless communication. With this, the project can be made applicable in the medical sector involving speech abnormalities as well as human-computer interaction. The scope of the project can be further expanded through enhancements of virtual assistants such as Alexa, Google assistant and so on by embedding silent speech communication instead of voiced communication. Also systems can be developed to generate digital voices for people having trouble with speaking.

11. CONCLUSION

EMG-UKA Trial corpus was initially processed and the top ten most frequently uttered words were selected as labels. The time and frequency domain features were extracted and analyzed using various parameters such as amplitude, power and energy content. The extracted features were used to train the designed machine learning models.

A dual-channel circuit was designed based on the specifications required for the extraction of low power, low amplitude EMG signals. The signals were successfully extracted using the designed hardware. The signal was then processed and finally visualized in a graphical interface.

After being accessible to the OpenBCI hardware, EMG signals extracted from the facial muscles were transmitted to the Cyton board where all the signal processing activities were accomplished with fewer improvements with respect to the initial setup. The signal after processing was prepared as input for machine learning models and the models were trained for prediction of the uttered words. The predicted output of the model was then displayed on the terminal screen along with accuracy and loss.

Thus, Neuromuscular signals or EMG signals which are prominent within the frequency range of 1-100 Hz can be used to interact with a remote computer system. The project highlights the fact that human computer interaction using neuromuscular signals is feasible with a permissible error rate. The dependency on the physicality of the speaker, speaking rate and the operating environment added further complications in the project.

12. APPENDICES

A. Project Budget

Table 12-1: Budget of Purchased Items

S.N.	Title		Qty (pcs)	Rate (NRs.)	Price (NRs.)	Remarks
1	Ag-AgCl Electrode (ECG EMG)		150	10/-	1,500/-	-
2	Instrumentation amplifier (AD620)		5	114/-	570/-	-
3	Op-amp (OP37AJ)		10	114/-	1,140/-	-
4	Passive electronic components	• Resistors • Capacitors • Header pins • Diodes	-	-	2,000/-	-
5	AgCl Electrolyte		250ml	50/-	50/-	-
6	Arduino Uno		2	1,000/-	2,000/-	-
7	Single sided PCB board		2	250/-	500/-	-
8	Shielded RCA Cable		8 (1m)	215/-	1,720/-	-
9	Cyton Board		2	46,532/-	93,064/-	Donated
10	USB Dongle		2	11,633/-	23,266/-	Donated
11	Gold Cup Electrodes		2 set	3,490/-	6,980/-	Donated
12	Ten 20 Electrolyte		6	2,326/-	13,956	Donated
13	Miscellaneous		-	-	3,000/-	-
	Total				**1,60,212/-**	

B. Project Timeline

Table 12-2: Gantt Chart

Project Start Date: 15 November 2019 Project End Date: 25 February 2021

Task	2019		2020												2021	
	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb
Brain Storming and Project Selection																
Research on Hardware Design																
Hardware Testing and Debugging																
EMG Signal Acquisition																
Dataset Preparation																
Feature Study																
Testing Machine Learning Models																
Model Implementation																
Model Testing in Terminal																
Documentation																

C. Module Specifications

Table 12-3: Specifications of Instrumentation Amplifier AD620

S.N.	Parameters	Specifications
1.	Gain Range	1-10,000
2.	Power Supply Range	±2.3 V to ±18 V
3.	Max. Supply current	1.3 mA
4.	Input Voltage Noise	0.28 µV p-p (0.1 Hz to 10 Hz)
5.	Bandwidth	120 KHz (G=100)
6.	CMRR	100 dB min (G=10)

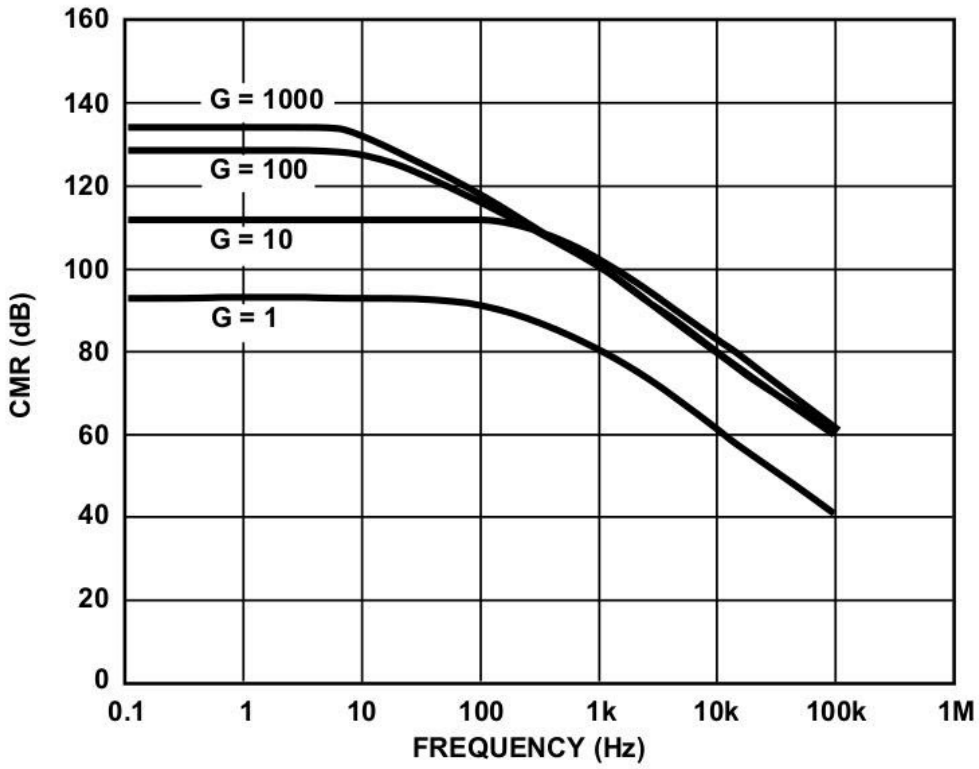

Figure 12-1: Typical CMRR vs Frequency Curve of AD620

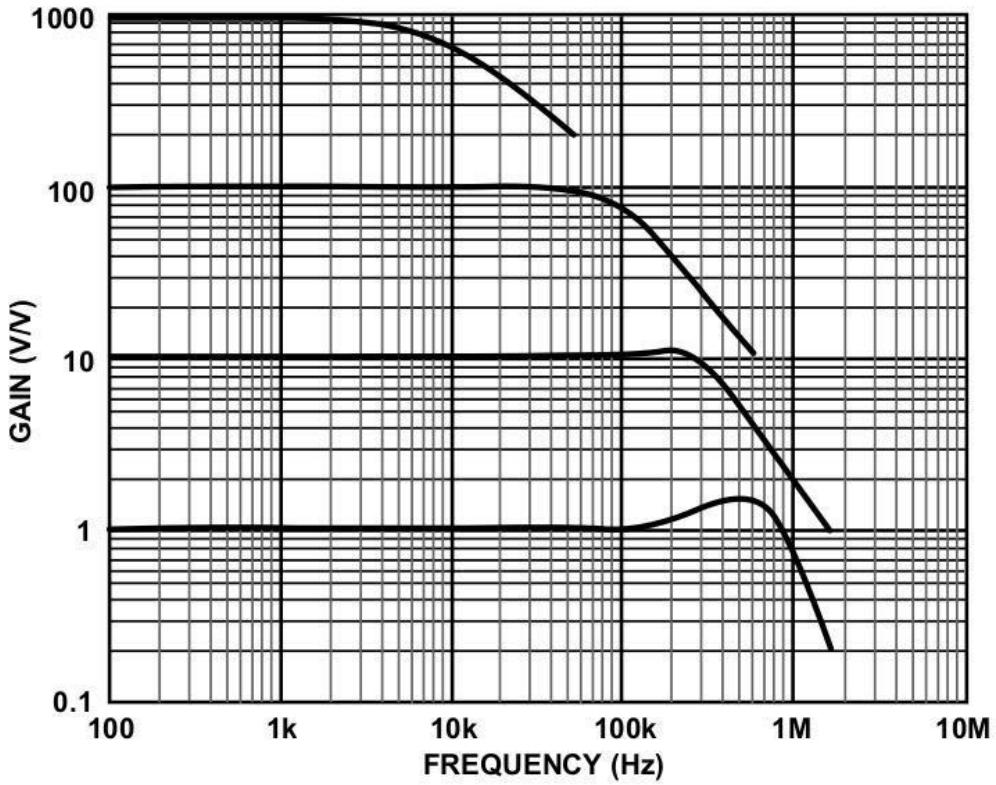

Figure 12-2: Voltage Gain vs Frequency Curve of AD620

Table 12-4: Specifications of Amplifier OP37G

S.N	Parameters	Specifications
1.	Open-Loop Gain	1.8 Million
2.	Max. Supply Voltage	22 V
3.	Max. Supply Current	25 mA
4.	Bandwidth	63 MHz (Common Voltage @ 11V)
5.	Input Voltage Noise	80 nV p-p (0.1 Hz to10 Hz)

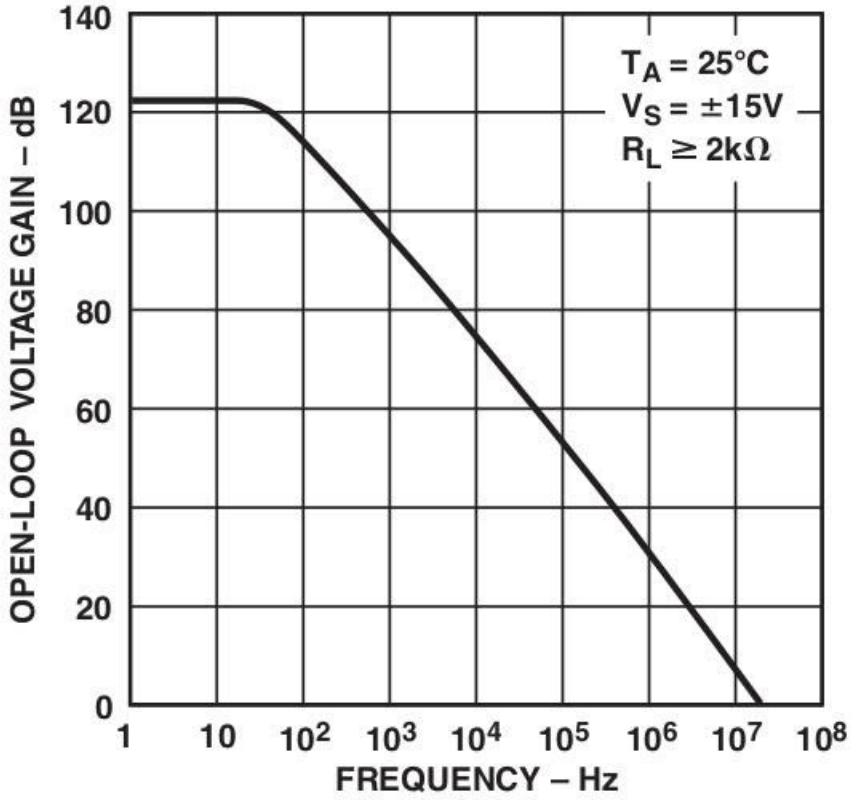

Figure 12-3: Frequency Response of OP37G

Table 12-5: Specifications of ADC of Arduino Uno

S.N	Parameters	Specifications
1.	Type	Successive Approximation Register
2.	Resolution	10 Bit
3.	Absolute Accuracy	±2 LSB
4.	Conversion Time	13 - 260μs

Table 12-6: Cyton Board Specification

S.N	Parameters	Specification
1.	Digital Operating Voltage	3.3 V
2.	Analog Operating Voltage	± 2.5 V
3.	Input Voltage	-3.3 to 12 V
4.	Resolution	24 bit
5.	Programmable Gain	1,2,4,8,12,24

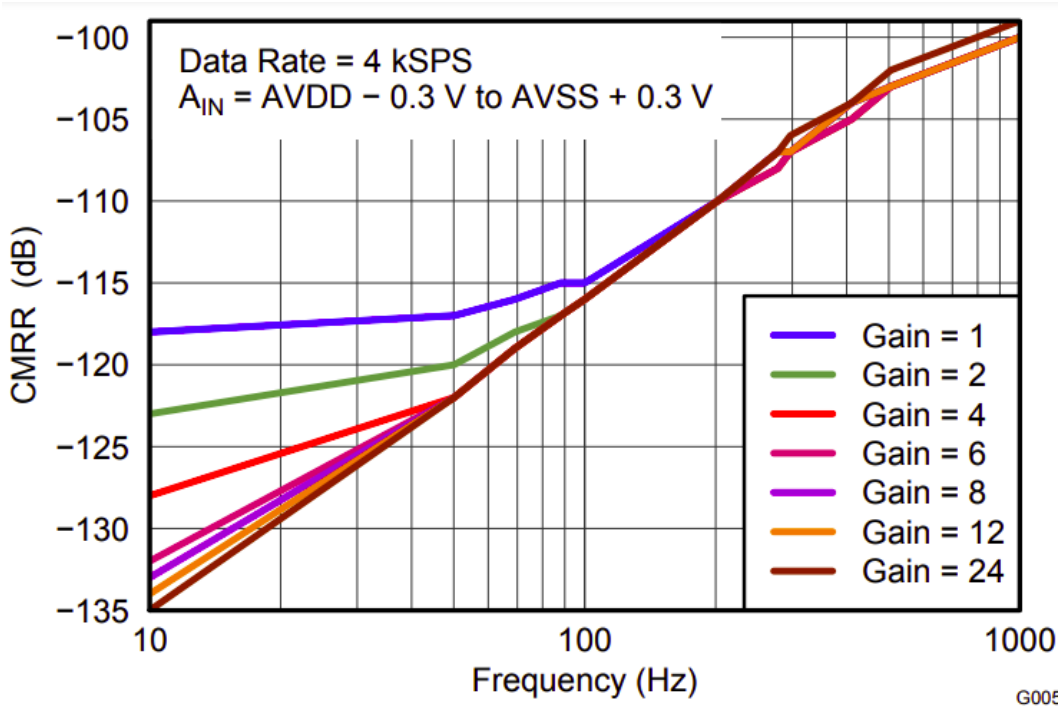

Figure 12-4: CMRR vs Frequency Curve of ADS 1299

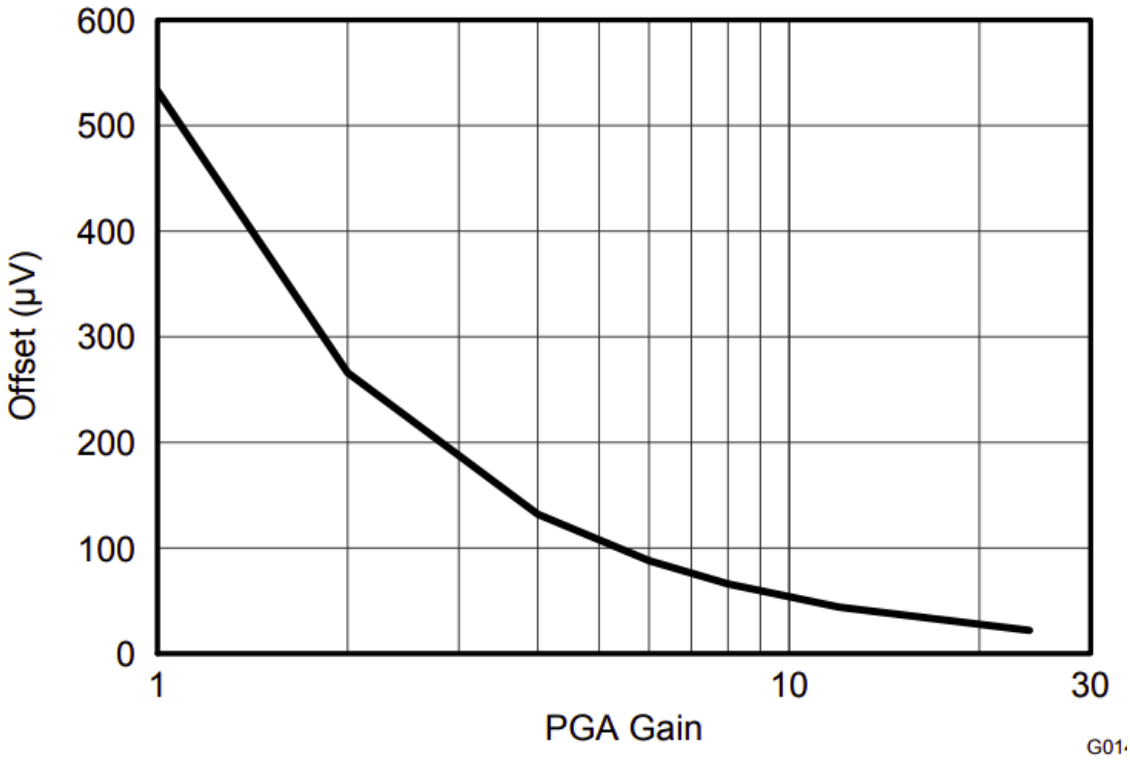

Figure 12-5: Offset vs PGA Gain (Absolute Value) Curve of ADS 1299

D. Raw Speech EMG Plot

EMG signal data of EMG-UKA Trial Corpus from different channels are as below:

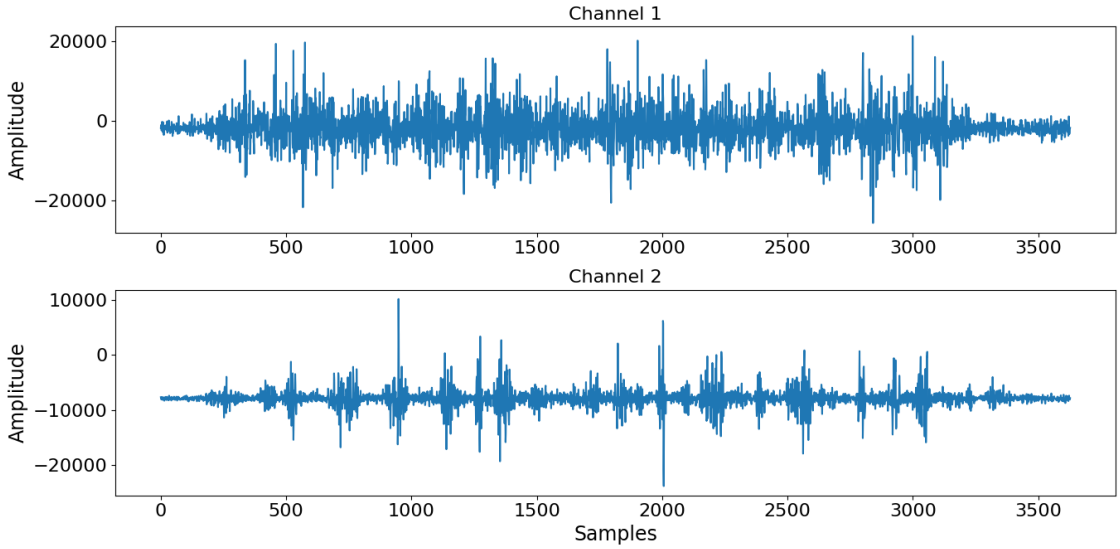

Figure 12-6: EMG Channel 1 and 2 From the Dataset

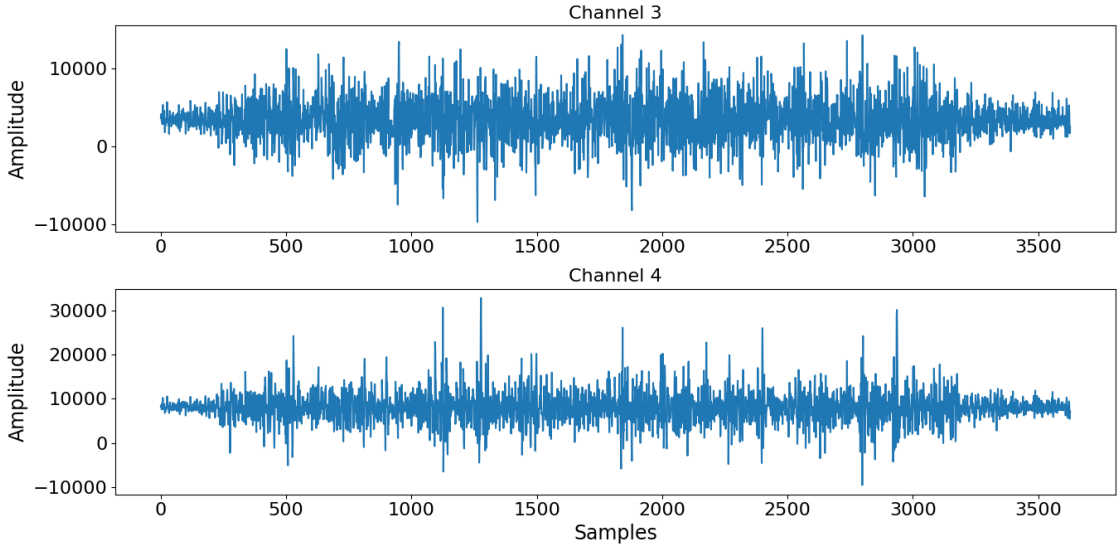

Figure 12-7: EMG Channel 3 and 4 From the Dataset

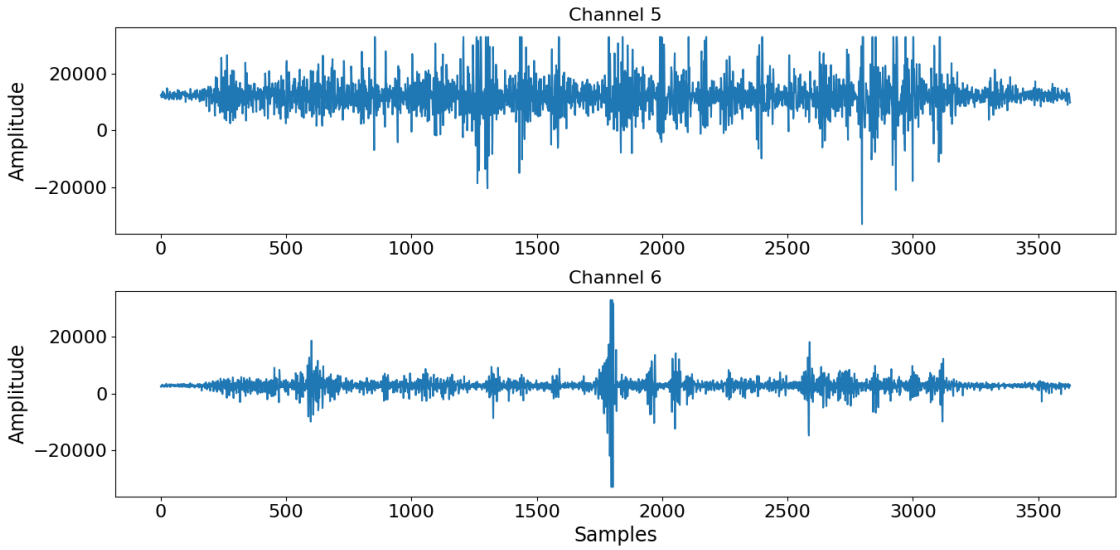

Figure 12-8: EMG Channel 5 and 6 From the Dataset

Above figures show EMG signals obtained from the dataset during citation of the sentence “THIS COUNTRY HAS RELIED ON IMMIGRANTS AND IS FOUNDED UPON A PRINCIPLE OF WELCOMING IMMIGRANTS” by the first speaker (profiled as “002”) during the first session. The figure 12-1 is the audio signal of the utterance and figures 12-2, 12-3 and 12-4 are the EMG signals of the utterance of the sentence from 6 different articulatory muscles as 6 different channels.

Above Signal is the raw EMG signal of the self-recorded dataset for the utterance of word ‘go’. The channel 1 - 8 is the signal from the 8 different articulatory muscles.

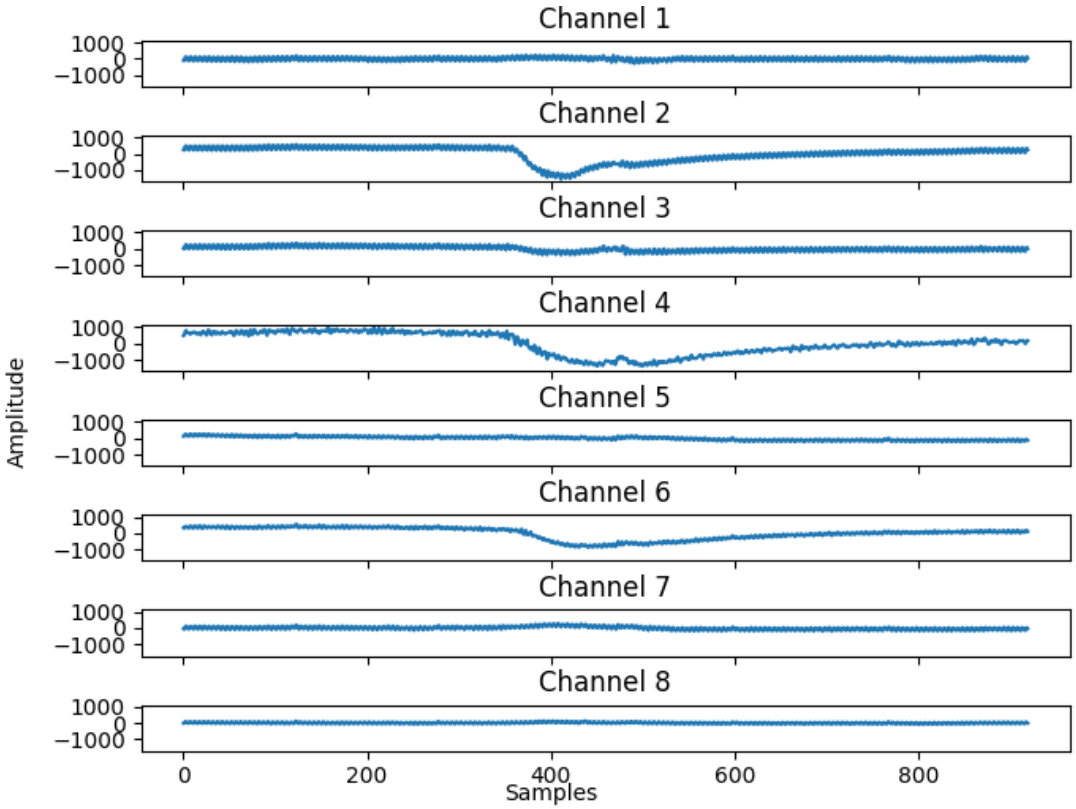

Figure 12-9: Self Recorded Raw EMG Signal

References

[1] A. Kapur, "Human-Machine Cognitive Coalescence through," MASSACHUSETTS INSTITUTE OF TECHNOLOGY, 2018.

[2] Arnav Kapur,Shreys Kapur, Pattie Maes, "AlterEgo: A Personalized Wearable Silent Speech Interface," *Multimodel Interface*, 2018.

[3] G. Gage, "Control Machines with your Brain," *Backyardbrains*, 2009-2017.

[4] Michael Wand and Tanja Schultz, "SESSION-INDEPENDENT EMG-BASED SPEECH RECOGNITION," *Cognitive Systems Lab, Karlsruhe Institute of Technology, Adenauerring 4, 76131 Karlsruhe, Germany*, pp. 1-3.

[5] Vanderthommen Marc, Duchateau Jacques, "Electrical Stimulation as a Modality to Improve Performance of the Neuromuscular System," *Exercise and Sport Sciences Reviews*, vol. 35, pp. 180-185, 2007.

[6] P. R. a. P. T. a. N. Agarwala, "End-to-end neural networks for subvocal speech recognition," 2017.

[7] M.Khan, M. Jahan, "The Application of AR Coefficients and Burg Method in Sub-vocal EMG Pattern Recognition," *Journal of Basic and Applied Engineering Research*, vol. 2, pp. 813-815, April-June 2015.

[8] Geoffrey S. Meltzner, James T. Heaton, "Development of sEMG sensors and algorithms for silent speech recognition," *Journal of Neural Engineering*, vol. 15, no. 4, 25 June 2018.

[9] Chuck Jorgensen and Kim Binsted, "Web Browser Control Using EMG Based Sub Vocal Speech Recognition," in *38th Hawaii International Conference on System Sciences, IEEE*, Hawaii, 2005.

[10] David Gaddy, Dan Klein, "Digital Voicing of Silent Speech," in *Conference on Empirical Methods in Natural Language Processing*, 2020.

[11] Jennifer C. Shieh, Matt Carter, Guide to Research Techniques in Neuroscience, Second Edition ed., Academic Press, 2015, p. 89.

[12] Arthur C. Guyton, John E. Hall, Textbook of Medical Physiology, Eleventh Edition ed., Elsevier Inc., 2006, pp. 57-91.

[13] K Sembulingam, Prema Sembulingam, Essentials of Medical Physiology, Sixth ed., Jaypee Brothers Medical Publishers (P) Ltd, 2012, pp. 197-199.

[14] "Phonation," [Online]. Available: https://www2.ims.uni-stuttgart.de/EGG/page6.htm. [Accessed 25 December 2019].

[15] M. Valkenburg, Analog Filter Design,, CBS College Publishing, 1982, pp. 157-167.

[16] "Electronics Tutorial," [Online]. Available: https://www.electronics-tutorials.ws/filter/second-order-filters.html. [Accessed DEcember 2020].

[17] Y.-S. C. Hyeon Kyu Lee, "Application of Continuous Wavelet Transform and Convolutional Neural Network in Decoding Motor Imagery Brain-Computer Interface," *Entropy (Basel).*, 5 Dec 2019.

[18] Y. Wang, "The Ricker wavelet and the Lambert W function," *Geophysical Journal International, Seismology,* pp. 111-115, 2015.

[19] Ulysse Cote-Allard, Evan Campbell, Angkoon Phinyomark, Francois Laviolette, Beniot Gosselin, Erik Scheme, "Interpreting Deep Learning Features for Myoelectric Control: A Comparison With Handcrafted Features," *Frontiers in Bioengineering and Biotechnology,* 2020.

[20] A. Mertins, Signal Analysis: Wavelets, Filter Banks, Time-Frequency Transforms and Applications,, 1999: John Wiley & Sons Ltd.

[21] Mariusz Kubanek , Janusz Bobulski and Joanna Kulawik, "A Method of Speech Coding for Speech Recognition Using a Convolutional Neural Network," *Symmetry*, vol. 11, p. 9, 19 09 2019.

[22] G. Kamen, David A. Gabriel, "Essentials of Electromyography," in *Human Kinetics*, 2010, pp. 57-71.

[23] "Ten20 Material Safety Datasheet," Weaver and Company, [Online]. Available: https://bio-medical.com/media/support/10208.pdf. [Accessed 27 January 2021].

[24] "OpenBCI," [Online]. Available: https://shop.openbci.com/collections/frontpage/products/cyton-biosensing-board-8-channel?variant=38958638542. [Accessed 16 December 2020].

[25] Michael Wand, Matthias Janke, Tanja Schultz, "The EMG-UKA Corpus for Electromyographic Speech Processing".

[26] "Understanding FFTs and Windowing," National Instruments, 05 03 2019. [Online]. Available: :https://www.ni.com/en-us/innovations/white-papers/06/understanding-ffts-and-windowing.html.

[27] "CS231n Convolutional Neural Networks for Visual Recognition," Stanford University, 2019.

[28] Ossama Abdel-Hamid, Abdel-rahman Mohamed, Hui Jiang, Li Deng, Gerald Penn, and Dong Yu, "Convolutional Neural Networks for Speech Recognition," *IEEE/ACM TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING,* vol. 22, no. 10, 10 2014.

[29] I. Analog Devices, "Low Cost, Low Power, Instrumentation Amplifier, AD620," Analog Devices, [Online]. Available: https://www.alldatasheet.com/datasheet-pdf/pdf/48090/AD/AD620.html. [Accessed 16 December 2019].

[30] D. J. Brownlee, "A Gentle Introduction to Cross-Entropy for Machine Learning," Machine Learning Mastery, [Online]. Available: https://machinelearningmastery.com/cross-entropy-for-machine-learning/.