Santa Clara University

Scholar Commons

Bioengineering Senior Theses   Engineering Senior Theses

6-2020

Brainwave Classification for EEG-based Neurofeedback

Brent Baculi

Stuart Cansdale

Follow this and additional works at: https://scholarcommons.scu.edu/bioe_senior

Part of the Biomedical Engineering and Bioengineering Commons

SANTA CLARA UNIVERSITY

Department of Bioengineering

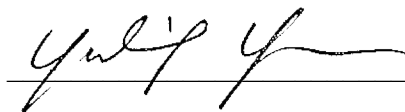I HEREBY RECOMMEND THAT THE THESIS PREPARED
UNDER MY SUPERVISION BY

Brent Baculi, Stuart Cansdale

ENTITLED

Brainwave Classification for EEG-based Neurofeedback

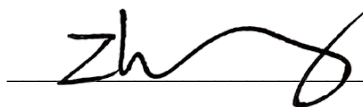BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
DEGREE OF

BACHELOR OF SCIENCE
IN
BIOENGINEERING

_____   6/9/20

Thesis Advisor   Date

_____   6/9/20

Department Chair   Date

Brainwave Classification for EEG-based Neurofeedback

Santa Clara University

Baculi, Brent and Cansdale, Stuart
Advisor: Yuling Yan, Ph.D.

Table of Contents

Abstract

The aim of this project was to find a way to differentiate active and rested brain signals in a patient using tasks without bodily movement to provide extremely motorly disabled patients a method of control for robotic devices that enable them to move independently of a caretaker. Although many control methods exist for less severely motorly impaired patients, this method would improve quality of life for all patients by allowing for movements to be controlled exclusively using the brain. The three steps for our project were to define the tasks and collect data, process the signals, and run the processed signals through a machine learning algorithm. In addition to the tasks not involving movement, having the subject's eyes open was required as closing one's eyes as a control method would not be practical. Different processing techniques were used to prepare the data and extract features for the training of the machine learning model for the classification task. Due to COVID-19, a limited amount of data was collected, resulting in inaccurate classification results. The "imagining-to-move" and "at rest" tasks that we designed for data collection appear to be the most effective when focusing on the mu rhythms at 7 to 12 Hz from the central cortex, but much more data is needed to prove this point. These tasks, brain area, and frequency ranges would be ideal for control method research projects in the future.

Acknowledgements

List of Figures

List of Tables

Chapter 1: Introduction

1.1 Background

Bipedalism, or the ability to walk on two legs, is an essential trait that we humans often take for granted when considering our quality of life. Having fully functional legs that can be manually controlled allows one to travel any distance they wish and participate in leisure activities involving leg movements such as biking and soccer. Unfortunately, those who have dealt with major brain or spinal cord injuries, progressive neural diseases, degenerative muscular diseases, and other debilitations cannot enjoy this human benefit.

According to the Christopher & Dana Reeve Foundation in 2013, approximately 5.4 million people are diagnosed with paralysis of some kind in the United States alone. Of these cases, stroke (30%); spinal cord injuries (27%) including motor vehicle accidents, sports accidents, and physical labor; and multiple sclerosis (18%) are the top three causes [1]. When singling out spinal cord injuries, approximately 291,000 persons suffer from spinal cord injuries with about 17,000 new cases of spinal cord injuries occurring every year, according to the National Spinal Cord Injury Statistical Center. After hospital discharge from spinal cord injuries, less than 1% of patients experienced complete neurological recovery [2]. The high frequency of spinal cord injuries and the likelihood of spinal cord injuries leading to any paraplegia warrants attention for a solution that can help those affected by paraplegia in their daily lives.

1.2 Problem

Physical therapy for paraplegic and tetraplegic patients is a necessary component to rebuild the communication between the brain and the spinal cord and strengthen the muscles that were severely weakened from the injury. These activities can range from leg and core exercises and gait training [3] to electrical stimulation therapy [4]. However, as with all physical therapy, complete rehabilitation, if even possible, takes a long time. While patients are undergoing the healing process, they are still subject to mundane tasks that are much harder to do because of their condition. This problem warrants a need for assistive devices that will aid the patient in physical tasks while the patient is undergoing physical therapy. By incorporating these medical devices, paralyzed patients will have a greater range of motion, allowing them to complete daily tasks easier and gain back the confidence to do these tasks independently.

1.3 Current Methods/Research

1.3.1 Electroencephalography (EEG)

The electrical signals, or brain waves, that are outputted by the neurons in the cortex can vary depending on the mental state of the person. These brain waves can be recorded and captured using Electroencephalography (EEG). Flat metal disks, or electrodes, are attached to the scalp in such a way that they record the activity of each lobe of the cortex [5]. The internationally recognized way of placing these electrodes is through the 10-20 system, a map of the ideal placement of electrodes based on the four lobes of the cortex. These brain waves are delivered as signals to a computer which then analyzes the frequencies, or the amount of waves that are repeated over a given time, of these waves.
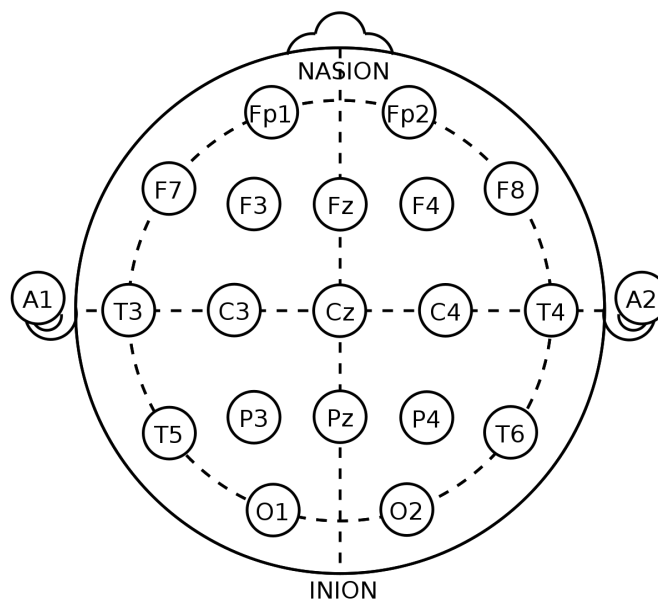
Figure 1.1 The International 10-20 System Map. The electrodes are placed such that the brain activity of every lobe can be captured (Google Images).

By default, the EEG records the entire range of frequencies that are emitted from the brain, including the noisy, extraneous artifacts. By implementing preprocessing and postprocessing steps, or filtering the data before and after recording brain wave data respectively, one can achieve a clearer wave. After filtering, one can visualize one of five basic patterns of brain waves that are either suppressed or apparent based on the mental state of the subject: Delta, Theta, Alpha, Beta, and Gamma. The frequency ranges and the typical behavior of the status in which these waves are active are as follows [6]:

- Delta (0 - 4 Hz): Deeply Asleep
- Theta (4 - 7 Hz): Light sleep, meditation, visual imagery
- Alpha (8 - 13 Hz): Awake, relaxed

- Beta (14 - 30 Hz): Alert, concentrated
- Gamma (> 30 Hz): Peak focus

A variant of alpha waves called mu waves are thought to originate only in the motor cortex of the brain. These waves are suppressed when the subject is performing a movement. Conversely, mu waves are apparent when the subject is at rest but awake. It is theorized that these mu waves are similarly suppressed when one plans to do a movement but does not actually move [7]. This theory will be a key component to the construction of our machine learning model.

The electrodes can be placed in two ways: invasively (electrodes are required to be surgically implanted in the brain) and non-invasively (electrodes are placed on the anterior of the scalp). While invasive EEG methods may have higher transfer rates in the electrodes, there are multiple financial and ethical costs that are simply not worth the increased rate [10, 9] Noninvasive EEG methods are preferred to reduce as much risk as possible, but because of the distance away from the brain, these EEG recordings are more susceptible to noise from artifacts and are more limited in communication with brain waves than invasive EEGs [9]. Nevertheless, many experiments have been executed to optimize the non-invasive EEG method.

EEG is primarily used as a diagnostic tool. One of the most notable diseases that EEG can diagnose is epilepsy, in which the patient suffers from uncontrollable electrical activity in the cortex, causing a seizure. Using photostimulation to trigger the brain to activate the epileptic potentials, the EEG can record these potentials to diagnose the patient's brain wave pattern [6]. However, a multitude of technologies have incorporated EEGs by developing a brain-machine interface (BMI), a technology that allows interaction between neuronal tissue and artificial devices, thereby serving as a control method [9].

1.3.2 Physical Therapy for Paraplegic Patients

1.3.2.1 Gait Training

Physical therapy is the most thought-of method when dealing with anatomy that must be strengthened to regain ability. One of the most common ways of strengthening damaged parts of the spinal cord is **gait training**. This form of therapy often uses a combination of robotic-assisted devices and body weight support to condition the legs to walk further without tiring. In a study to evaluate the effectiveness of weight-supported gait training through literature review, Wessels et. al. in 2010 discovered that subjects under a one year gait training program performed better in the Functional Independence Measure compared to those who underwent treadmill-based training [15]. This comparison is significant because gait training demonstrates a more reliable approach to restore independent movement. However, one evident shortcoming of gait training is that patients will not have immediate control of their damaged anatomy. Training usually takes a year to

notice significant improvements in the damaged area. Until then, patients will be required to rely on other able-bodied persons to assist them in everyday tasks.

1.3.3 Advancements in Brain-Machine Interfaces:

1.3.3.1 EEG-based Control System for a Robotic Arm:
      Despite being a relatively new field, the field of BMIs have seen some major advancements. In an attempt to create an **EEG-based control system for a robotic arm**, Yoshioka et. al. devised an experiment in 2014 that extracted mu and beta rhythms from the motor cortex's neurons using an electro-cap and using short-time fourier transform (STFT) to analyze the data. Yoshioka assigned four tasks for the patient to accomplish: idling, gazing, imagining motion, and doing a motion. Expectedly, the spectrogram from the STFT showed that the mu waves were the highest when the subject was idle and were suppressed during the other three tasks. Conversely, the beta rhythms were shown to increase from when the subject is idle to when the subject is doing a motion. They concluded that the imagining task did not demonstrate the ideal, theoretical results they were looking for because the subject could not concentrate for more than ten seconds [9]. In terms of this study, the tasks of moving the subject's arms and imagining said movement was convenient because of their ultimate goal of moving a robotic hand. Observing the spectrograms of the different tasks, there are noticeable changes in the frequency ranges. However, because the subject could not easily maintain a consistent concentration for one task, the data ends up being choppy, making it hard for the robotic arm to read the data properly.

1.3.3.2 Pictographic Writing System:
      In 2018, Stach, Browarska, and Kawala-Janik, faculty members of the Electrical Engineering department at Opole University of Technology, tested the **Emotiv EPOC**+, a relatively inexpensive mobile EEG headset, by creating a **Graphical User Interface intended as a pictographic writing system**. Using the Emotiv headset, the patient would answer a question by hovering over an option using the mouse cursor and then blinking to confirm the answer. Even during its initial phase, the authors inferred that the software could not conform to people with other various disabilities [11]. By having a universal EEG system that can assist a patient regardless of disability, this interface would be more accessible.

1.3.3.3 Noninvasive Neuroimaging for Enhanced Neural Tracking:
      In 2019, B.J. Edelman and other researchers including Bin He, the department head of Bioengineering at Carnegie Mellon University, recognized that recorded EEG data is subject to much noise even after preprocessing and postprocessing methods. To further improve the EEG signal quality, they designed a framework that **enhances neural tracking for robotic device control** by using noninvasive neuroimaging to further filter out noise

[13]. This framework would especially be helpful to the robotic arm mentioned previously as it would account for a human's lack of constant state of mind. However, while the framework is great for further filtration, their experimental design heavily relied on able-bodied movement. Regardless, because of the noise one frequently encounters in noninvasive EEG data, this framework can serve as a building block to a future EEG controlled device.

1.3.3.4 *Brain-computer Interface Wheelchair*

Alternatively, EEGs have been used to move motorized devices to provide another method to increase range of movement. In 2011, Khare et. al. [12] attempted to create a **brain-computer interface (BCI) that could assist severely disabled persons move a wheelchair**. Khare attached the electrodes near the motor cortex, the frontal lobe, and the parietal lobe to access the neurons attuned for motor movement. To control the wheelchair, instead of having the subject think about which direction the wheelchair should go, he assigned tasks for each direction the wheelchair would go. For example, two of the tasks presented were at rest, which coincided with stopping the wheelchair, and trivial multiplication (ex. 1 * 2 = 2), which coincided with going forward. This novel approach alleviates the subject's strain of imaging an object to move a certain direction, thereby allowing the subject's brain to be more constantly in a state of mind. However, a notable limitation is that the purpose of these tasks may not be as evident to the subject. If the subject was told to make the wheelchair go forward, they may put less emphasis on the task to do so and more of wanting to make the wheelchair push forward.

1.3.3.5 *Wheelchair with Graphical Interface:*

In a similar fashion, another experiment by Iturrate et. al. (2009) offered a **graphical interface of the surrounding area** that the subject can focus on instead of mentally stimulating tasks. When there is visual stimulation, the EEG detects the location which is transferred to an automated navigation system that takes the subject to that location without colliding into other obstacles. Allowing the subject to complete a more stimulating task rather than simply imagining an action will result in more unique potentials firing in the brain [14]. If the subject becomes too distracted in their tasks, they may revert back to the rested mental state unknowingly.

1.4 Goal and Significance

The goal of this project is to train a machine learning model that can differentiate between brain activities when the subject is at rest (i.e. not moving any part of the body) or focusing on a task. By differentiating between the various states of the brain, one can use this data to initiate commands to a machine to perform a specific task. This model could lend itself to assistive devices with the purpose of improving the quality of life for motor-impaired patients. Consequently, the users of these devices would enjoy more independence, more confidence, and greater well-being and conscience.

Chapter 2: Project and Systems Overview

In order to fulfill this goal, we imposed a very important restriction on our design: to introduce tasks that wouldn't involve any motor function. There are many control methods that exist already, such as voice commands, hand controls, and pneumatic mouth controls. However, all of these control methods involve motor tasks and don't help severely impaired patients. Not only does our design restriction include motor-disabled people under our design scope, but it would also improve the ease of access for other patients as well. Without needing to roll a wheelchair, a patient has access to their hands. Without needing to issue voice commands, a patient is able to speak freely. The ideal far future of this technology is to introduce control methods to the disabled to allow them to move around purely using their brain as if they were abled.

2.1 Project Components

Our project requires three main components: a device to collect data, a way to process data, and a way to analyze data. We chose to collect our own data instead of searching for an online data set in order to impose restrictions on the tasks performed when collecting data as well as having full control over which areas of the brain we would focus on. A way to process data is necessary for improving the accuracy of the analysis we perform. EEG data is very noisy, and it's also hard for patients to stay focused. That is why it is vital to process data using transforms and filters. These processing methods were done using AcqKnowledge, MATLAB, and Python. Finally, our project needs a way to analyze the processed data for feature extraction. The goal of the project is to acquire and prepare our EEG data sets effectively so that we can train a machine learning algorithm to classify different brain states with a high accuracy. This part would also be done in MATLAB and Python.

2.1.1 Task Creation

As aforementioned, we designed specific mental tasks for our project to offer control with an ultimate goal to assist the severely motorly impaired. There are many options to offer aid to the motorly impaired, but very few options for those who suffer from severe motor disabilities. Therefore, whenever we created tasks, we had the following requirements in mind:
1. The task must not involve muscle movement.
2. After the tasks are completed, there must be a noticeable difference in brain activity.

We initially started with multiple focusing tasks. This included advancing the eye forward on a checkered surface, switching between focusing on a black up arrow versus a white down arrow, and focusing on points far away from the patient. After attempting

many of these tasks, we decided to move more in the direction of focusing to move specific parts of the body without moving said body parts. To do this, we came up with three different tests:

1. Have the patient **relax** in a chair with their eyes moving while not moving. This was to simulate a patient that would have an active mind but motorly impaired to where they couldn't move.
2. Have the patient **move their arm up and down at the elbow joint**. This test acted as a control method to compare to the third test.
3. Have the patient sit in the chair without moving and their eyes open while **imagining themselves performing the task from the second test**. This set of tests showed the most promise when conducting the initial post-processing methods, so we conducted the rest of our project using this data.

2.1.2 BIOPAC

EEG data was collected using a BIOPAC system with an ECI electro-cap and built-in electrodes that enable multi-channel data acquisitions. In particular, we had 8 different available channels that we were able to connect to 8 of the electrodes of our choice on the cap. The data was captured using the Acqknowledge software that came with the equipment. The setup is described in Section 3.2.

2.1.3 MATLAB and Python

We used MATLAB and Python to analyze the EEG data from the BIOPAC system. The purpose for the analysis is threefold:
1. Filter out any noise that AcqKnowledge's innate preprocessing algorithm did not catch.
2. Quantify the EEG data for the machine learning model for easier classification. This included transforming the data by Fast-Fourier Transform, Short-Time Fourier Transform, and Power Spectrum Density, and taking the root-mean-square of the EEG data.
3. Train an SVM model to classify a set of data given to it under two different classes: active and resting state.

2.1.4 Robotic Components (Proof of Concept)

The created model was going to be implemented to move a prototype wheelchair or toy car. We planned to reprogram a prebuilt remote control car and use the filtered EEG data to move the remote control car. However, due to complications caused by software issues and COVID-19, the prototype was unable to be built.

2.2 Master Materials List

1. Laptop with AcqKnowledge software
2. MATLAB
3. Python
4. AcqKnowledge dongle (USB)
5. MP160 AcqKnowledge Hardware
6. Amplifier Input Model #AMI1000D
7. 8 EEG100C Amplifiers
8. 8 5" 1.5mm Stackable Jumper Cable # JUMP100C
9. 10/20 Brain Map
10. 8 3m Mod Extension Cable, C-series TP # MEC110C
11. ECI Electro-Cap (CAP-SMALL, CAP-MEDIUM, CAP-LARGE))
12. ECI Electro-Cap Instruction Manual
13. Electrode Board Adapter Connector (rainbow cable)
14. 2 E6 Disposable Sponge Disks
15. E5-9S: Ear Electrodes - 9MM-SOCKETS
16. "Y" CableCBL204
17. Electrode Abrading Pad # ELPAD
18. E3-M: Body Harness-Medium
19. E12 Head Measuring Tape
20. BD 5 ml Syringe with Luer-Lok Tip
21. BD PrecisionGlide Needle 1.6mm x 19mm
22. ECI Electro-Gel
23. USB 2.0 10/100 Mbps Ethernet Adapter - Black
24. ACCORD-HUS Power Cord # AC150A
25. Ivory Dishwashing Soap
26. Office Chair with Chair Tilt Function
27. Cushion or Neck Support (optional)

2.3 Budget

The predicted budget for this project was detailed in **Table 2.1**. However, because we did not get to the prototyping stage, we did not dip into our allotted budget. **Table 2.1** is meant to show the items we would have purchased if we did reach that stage.

Item	Budgeted Amount
Network Data License	$1000
Impedance checker	$500
Arduino Kit x2	$50
Programmable Toy Car	$20

Car Parts (Motors, Transistors, Chip sockets, etc.)	$50
Total	**$1620**

Table 2.1: List of Budgeted Items. Amounts budgeted to items that were planned to be used in our experiment. The total allotted amount given to us is $300.

2.4 Timeline

The timeline for this project is detailed in Figure 2.1. This project took place during the Fall 2019, Winter 2020, and Spring 2020 quarters each with their respective deadlines and deliverables. There are four stages of this project, of which the first three have been reached:
1. Setting up the software and hardware (Fall 2019)
2. Collecting data from the tasks that were created (Winter 2020)
3. Creating a SVM model that can differentiate between EEG data (Spring 2020)
4. Creating a prototype wheelchair that implements the SVM model. (Not reached)

Chapter 3: Data Processing

3.1 Introduction

To determine the ideal electrodes and filtration parameters for our project, we designed a system that filters brainwave data as it is being collected at the areas of interest. An initial set of electrodes was chosen (see **Figure 1.1)** based on their proximity to the motor cortex, the area of the brain that typically demonstrates the most activity when one performs any motor action. The chosen electrodes and primary purpose for collection are defined in **Figure 3.1 (below)**.
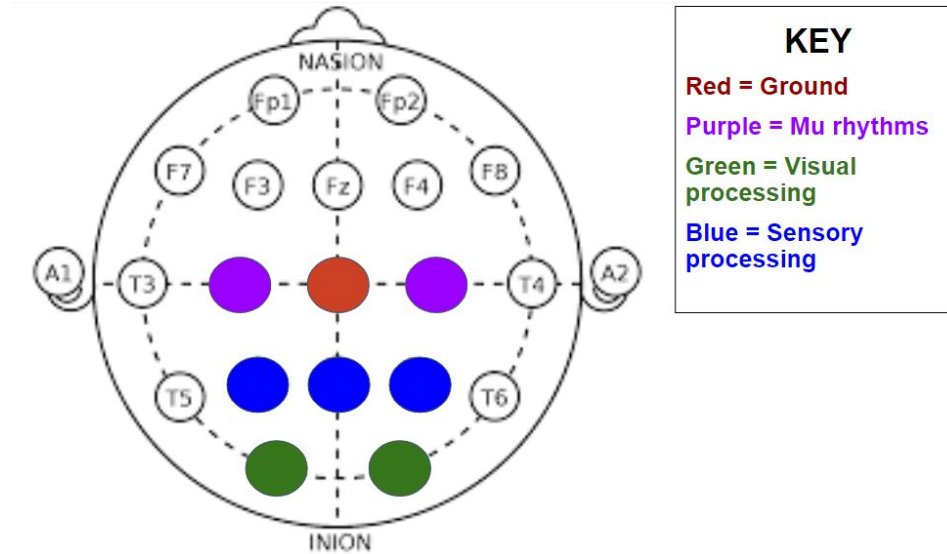
Figure 3.1: Areas of Interest for Data Collection. Electrode Cz (in Red at immediate center) was chosen as a reference point to compare the other surrounding electrodes. Electrodes C3 and C4 (in purple left and right from Cz) were chosen primarily for mu rhythm extraction. Electrodes P3, Pz, and P4 (in blue directly below the central cortex electrodes) and O1 and O2 (in green directly below the parietal electrodes) were analyzed to observe the brainwaves of those regions as the subject was performing various tasks. Electrodes A1 and A2 (on the ears, not highlighted) were attached using the ear electrodes for impedance.

The BIOPAC system can collect up to sixteen analog channels of raw EEG data simultaneously. Of these sixteen channels, eight were used to collect raw data from the electrodes mentioned above. Sixteen additional calculation channels were set up to pre-process the data, eight of which reduced the sampling rate to 125 Hz while the other eight took the 125 Hz channels and set a band-pass filter from 7.5 to 30 Hz to derive the alpha, mu, and beta waves from each of the EEG signals. The latter eight channels were then going to be compared against each other to determine the optimal electrode locations to observe, but due to time constraints, the electrode C3, was the only data manipulated after extraction and filtration.

3.2 AcqKnowledge Setup Procedure

1. Make sure AcqKnowledge is installed and the AcqKnowledge USB dongle is attached to the computer.
2. Attach the eight EEG100C Amplifiers to the AMI1000D. Then attach the AMI1000D to the MP160 AcqKnowledge Hardware.
3. Assign each amplifier to an analog channel using the switches on the top of the amplifiers (preferably 1 to 8)
4. Plug the MEC110C cables to the front of the EEG100C amplifiers
5. Flatten and secure the cables on a table to prevent tangling and order the cables in such a way that one can match the cords with their respective amplifiers

6. Attach one end of a JUMP100C to V- on cable 1 for the CZ electrode. Attach the other end of the JUMP100C cable to the ear electrodes
7. Connect the V- to every amplifier by plugging every successive V- into the previous one (e.g. plugging amplifier 2's V- into 1's JUMP100C socket and amplifier 3's V- into 2's JUMP100C socket)
8. Plug in the tip for each chosen electrode from the Electro-Cap into the V+ section of the respective amplifier chord. Refer to page 6 of the Instruction Manual to find the coordinating cables for each electrode.
9. Label each amplifier as well as the tips of the chord with their respective electrode so as to not confuse the channels
10. Plug in the power cord to the MP160
11. Plug in the ethernet adapter to the Biopac hardware and its USB to a laptop
12. Turn on the computer and open the AcqKnowledge software
13. Turn on the MP160.

3.3 Data Acquisition Procedure

3.3.1 Preparing the Subject for EEG Data Collection

1. Line the edge of the blank end of the color-coded tape measure provided by BIOPAC with the nasion. Wrap the tape measure around until the edge intersects with a color on the other end of the tape. This color corresponds to the electro cap that the subject will wear during data collection. If the edge falls in between colors, use the smaller option.
2. Tightly strap the appropriate harness under the subject's armpits making sure that the four sockets are in front of the patient and centered.
3. Gently abrade the earlobes using the Abrading Material. Clip the ear electrodes onto the subject's earlobes. It does not matter which electrode attaches to which ear. Make sure the metal part of the electrodes are at the front.
4. Fill the syringe with 5 mL of electro-gel and attach the PrecisionGlide needle. Using the syringe, poke the needle into an electrode hole and slowly fill it with gel. Repeat for the other ear.
5. Peel off the paper on the back of the sponge disks and attach the adhesive side of the disk onto the Fp1 and Fp2 electrodes from the inside of the proper sized cap (Refer to the 10/20 Brain Map to find the Fp1 and Fp2 electrodes).
6. Measure the nasion (the divet just above the nose) to inion (the bump on the back of the skull) with a centimeter tape. Divide that result by 10. Then, measure the distance up from the nasion and mark the derived distance with a marker.
7. Align the holes in the Fp1 and Fp2 mounts on the forehead mark. With the fingers on the inside and the thumb on the outside, "anchor" Fp1 and Fp2 on the forehead and start to slip the cap onto the head from the front to the back in a smooth motion.

8. Attach the ball on the left strap of the cap to the second socket from the right on the harness. Then, attach the ball on the right strap of the cap to the second socket from the left on the harness. The straps should cross.
9. Verify that the cap is stretched over the head and under constant tension. If the cap is not on tightly, numerous artifacts will result.
10. Using the 10-20 Map and the syringe, poke the needle into the cap electrodes of interest and slowly fill the electrodes with gel.
11. Connect the cap connector to the electrode board adapter connector attached to the MP160 hardware.
12. Turn on the computer with the AcqKnowledge software
13. Turn on the Biopac hardware
14. Open AcqKnowledge

3.3.2 Setting Up Channels for Pre-processing

The goal of the pre-processing channels were to do two things: to set up an anti-aliasing low-pass filter and to create a band-pass filter. This procedure details how we set up those two filters. The low-pass filter would be set at 125 Hz because we chose to divide the sampling rate of 2000 Hz by 16. We could have divided it by 32 for a low-pass filter of 62.5 Hz because the frequency range of focus is 7.5 to 12.5 Hz. Instead, we added a second band-pass filter of 7.5 Hz to 30 Hz. This setup allows us to add higher frequencies to our dataset if need be while still performing the necessary pre-processing of the EEG signals.

1. On the AcqKnowledge Software, go to MP160 > Set Up Data Analysis
2. Under the "Analog" Tab, connect the channels to their respective MP160 amplifier
3. Name the analog channels according to their respective electrodes
4. Under the "Calculation" Tab, create 8 channels that will be used for anti-aliasing each channel. To make one channel, click on "Filter - IIR" on the Preset drop down-menu, and leave the Channel Sampling Rate at 2000 Hz. Then, go to the "Setup…" button on the top right. On the "Source:" drop-down, click on the respective analog channel (A#, (title of analog channel)). On the "Output:" menu, click on "Low Pass". Then, set the Frequency to "Sampling rate / 16". Repeat for the next seven channels.
5. Next, create 8 more channels that will be used to preprocess data. To make one channel, click on "Filter - IIR" on the Preset drop down-menu, and leave the Channel Sampling Rate at 2000 Hz. Then, go to the "Setup…" button on the top right. On the "Source:" drop-down, click on the respective Calculation channel (C0-7, (title of calculation channel)). On the "Output:" drop-down, click on "Band Pass Low + High". Then, set the low frequency at a fixed 7.5 Hz and the high frequency at a fixed 30 Hz. Repeat for the next seven channels.

6. At this point, there should be 24 channels: 8 analog and 16 calculation. When running an EEG experiment, AcqKnowledge will run and display all 24 channels.
7. Save this file as "TemplateX.Y". If any changes need to be made during the experiment, repeat any of the steps above as necessary and save it as another file. Make sure to use your most recent template while keeping track of old templates.

3.3.3 Recording Data

The goal of this section is to test the tasks that we designed (see **Section 2.4**). Based on the tasks being performed, steps 4 to 8 are subject to change. For the tasks idling, imagining, and waving, the procedure is listed below:

1. This experiment requires *at minimum* 2 persons: one <u>subject</u> and the tester(s). Make sure that the subject is properly equipped and the tester(s) have control of the Acqknowledge software and can easily take note of any significant events during recording.
2. Instruct the subject to sit upright on the chair and permit them to lean back. Use a cushion or neck support if necessary to minimize head movement.
3. Open your most recent template file with your desired parameters.
4. Instruct the subject to relax their mind and keep their eyes open without moving any other body part until notified by a tester. After about 5 seconds, press the "Start" button at the top left area to begin recording EEG data. Leave the software running for about 30-60 seconds, making note of any significant events by either pressing the "Flag" icon at that point or writing it down with a timestamp.
5. When finished with the recording, instruct the patient to stop their activity. Then, press "Stop" where the "Start" button was at previously.
6. Save the file by going to "Save As" and rename the file as appropriate.
7. Instruct the subject to wave their right hand while still sitting upright, relaxed, having eyes open, and without any other muscle movement. Repeat Steps 4 through 6.
8. Instruct the subject to think about planning to wave the same hand while still upright, relaxed, having eyes open, and without any other muscle movement. Repeat steps 4 through 6.

3.4 Clean Up Procedure

1. Save data and close AcqKnowledge software
2. Turn off the MP160 Hardware
3. Empty and clean the syringe
4. Unplug only the Electrode Board Adapter Connector (rainbow cable)
5. Unbutton the cap from the harness
6. Take off harness and cap

7. Wipe gel off of patient using gauze square
8. Unsnap and remove the straps that attach above the ears
9. Clean out ear electrodes with gauze or a cotton swab
10. Fill sink with lukewarm water and add a small amount of Ivory liquid detergent
11. Sling the rainbow cable over the shoulder. Submerge the cap only while keeping cable over the shoulder **making sure the cable does not get wet**.
12. Soak the cap for a few minutes
13. Clean gel from electrode mounts using rapidly running water
14. Rinse cap thoroughly to make sure all the gel is cleaned off
15. Hang up cap to dry and tape the rainbow cable above the cap to keep it from getting wet
16. Replace straps when the cap dries

3.5 Results

        Figures 3.2, 3.3, and 3.4 show the raw EEG data that was captured and the EEG data after filtering out extraneous artifacts caused by head movement and the metal electrodes themselves from each of the three tasks (see **Section 2.1.1**). EEG data is notorious for appearing random, but actually having some pattern when looking closely at the data. Even ignoring the fact that these artifacts existed, simply observing the raw data demonstrates that the data is not practical for classification due to its randomness.

        Since our experiment classified movement as our independent variable, we decided to hone on Channels 7 and 8 from each state, which recorded activity in the motor cortex. Interestingly, these channels generated the most intense waveforms compared to Channels 1 through 6. The waves of channel 7 on **Figure 3.2** were as intense as the waves at **Figures 3.3** and **3.4** despite the subject not moving at all. Additionally, the waves of channel 8 on **Figure 3.4** were less intense than those of **Figures 3.2** and **3.3**. These observations yielded a need to filter the data to better discern the waveforms.

        After filtering the waves to the specified frequency range, the channels have noticeably cleaner wave patterns. Conversely to the raw EEG data, the filtered data recorded similar wave patterns throughout every channel in every state. Obviously, we could not differentiate the wave patterns because of their apparent similarity. The similar shape of waveforms may suggest that waves between 7.5 – 30 Hz run similarly throughout the brain rather than one area having differently shaped waveforms with the same frequency range. Despite this observation, there do seem to be slight differences in the intensity of the waveforms. For example, at the 38 second mark of the filtered data in **Figure 3.2**, the peaks of Channel 7 and 8 are noticeably smaller than those of Channels 1 through 6. This observation begs the need to develop further processing techniques and calculations to extract those minute differences.

Figure 3.2: Two Seconds of Subject at Rest (Idle) with Eyes Open. Channels 1-8: Raw, unprocessed brainwave data. Channels 48-55: Data after applying anti-aliasing filter.

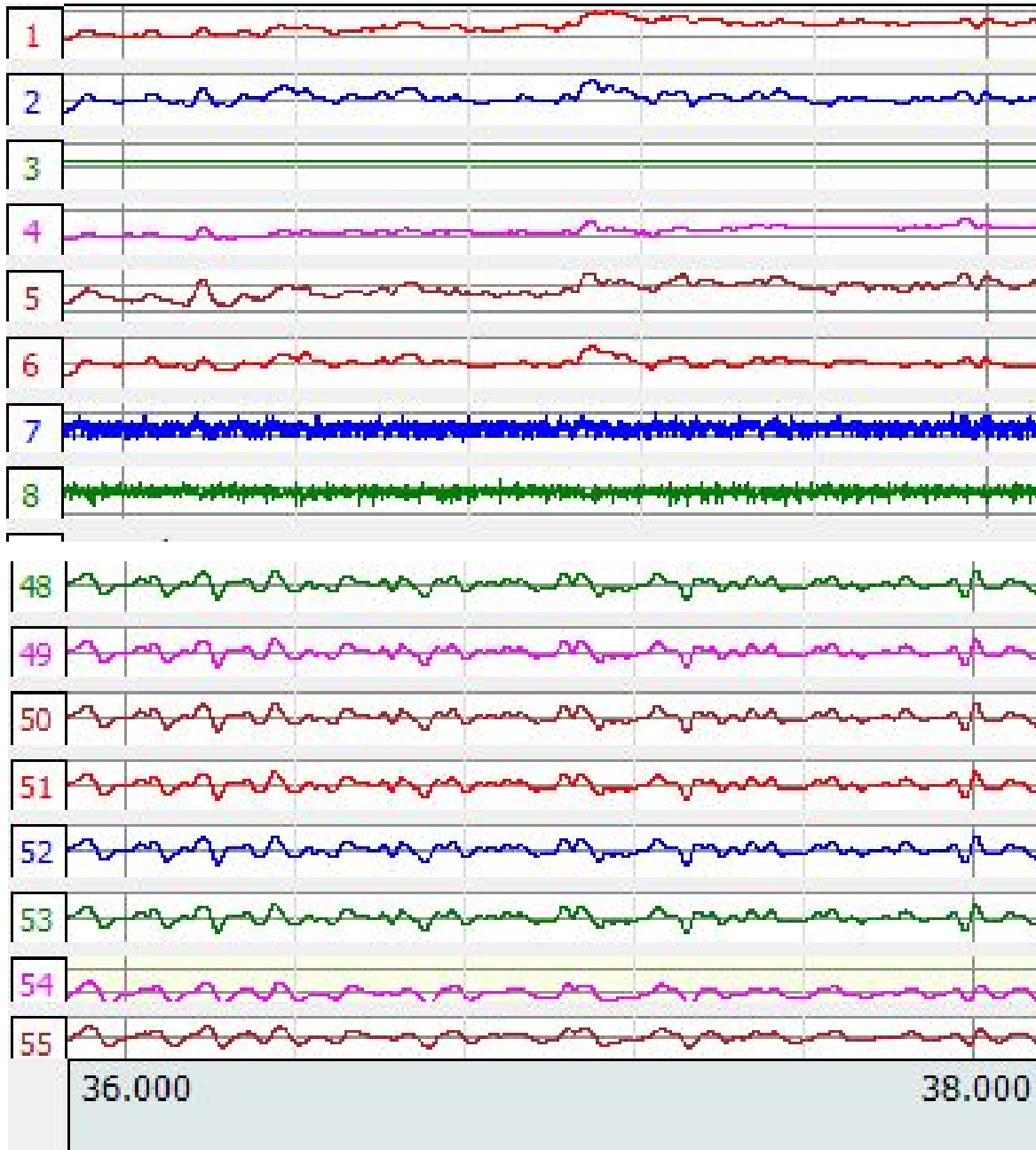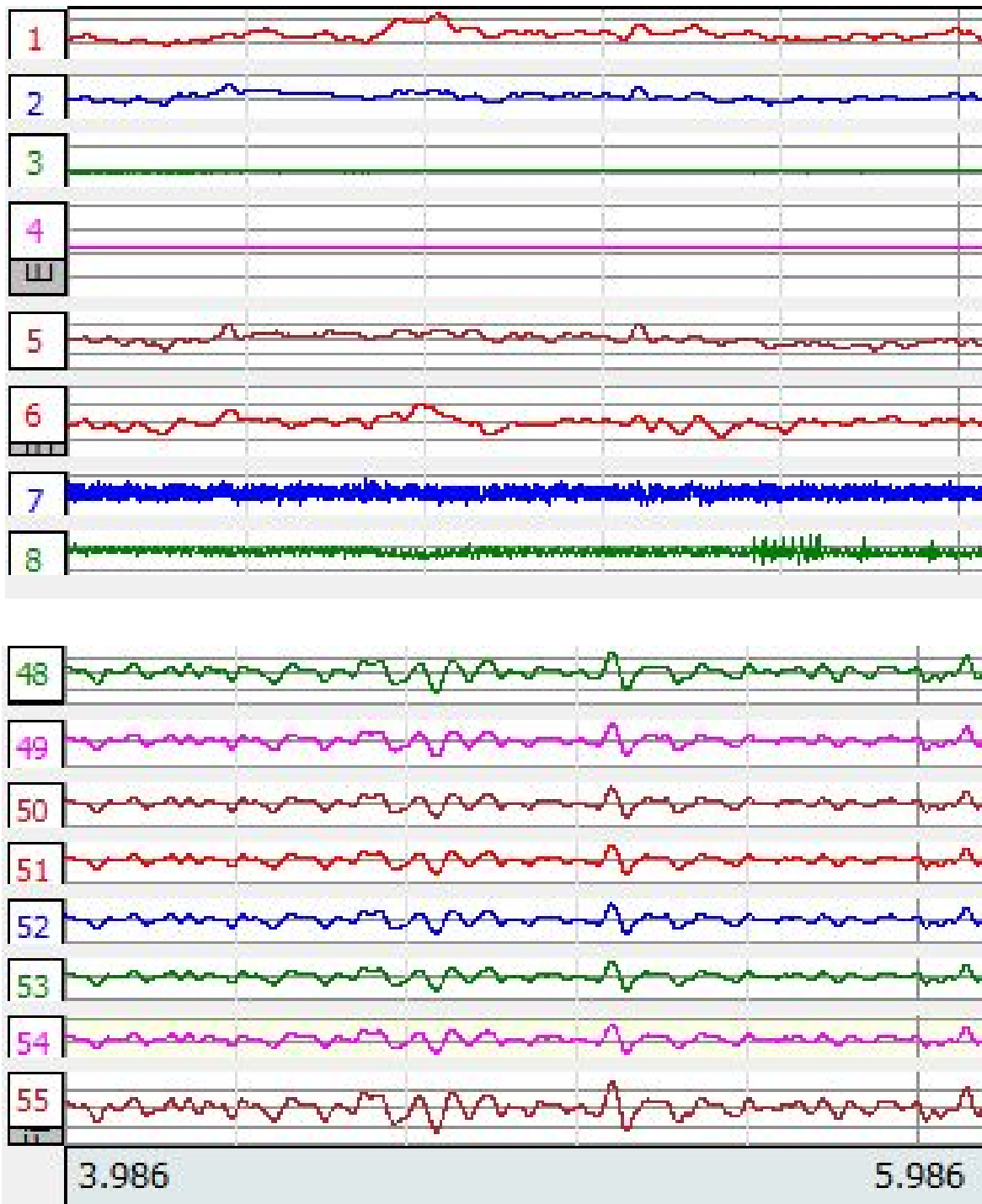**Figure 3.3: Two Seconds of Subject Waving Right Hand.** Channels 1-8: Raw, unprocessed brainwave data. Channels 48-55: Data after applying anti-aliasing filter.
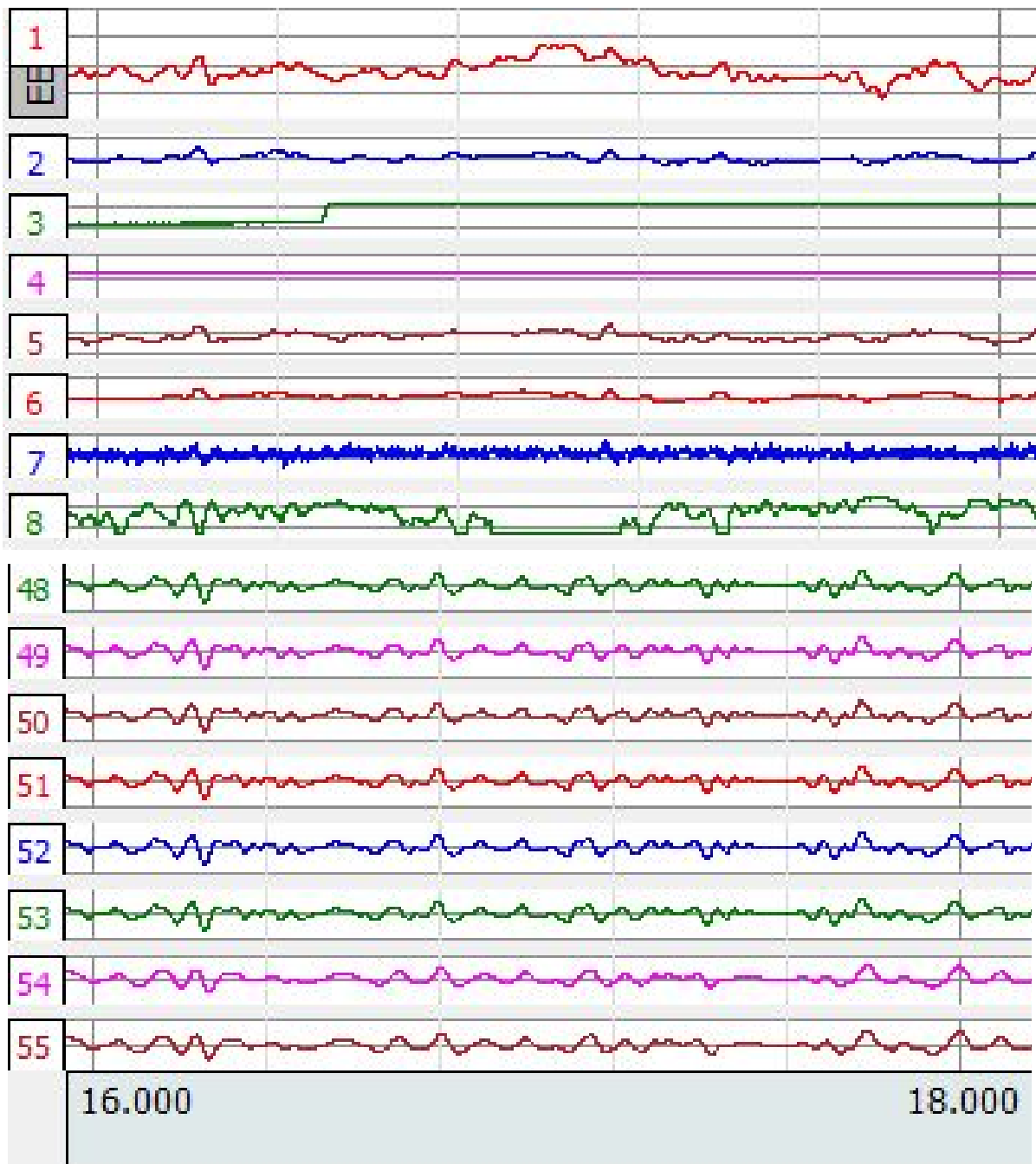
Figure 3.4: Two Seconds of Subject Imagining Waving Right Hand. Channels 1-8: Raw, unprocessed brainwave data. Channels 48-55: Data after applying anti-aliasing filter.

Electrodes	Channels	Region of brain
Cz	1, 48	Motor cortex
Pz	2, 49	Parietal lobe
O1	3, 50	Occipital lobe
O2	4, 51	Occipital lobe
P3	5, 52	Parietal lobe
P4	6, 53	Parietal lobe
C3	7, 54	Left Motor cortex
C4	8, 55	Right Motor cortex

Table 3.1: Legend to Figures 3.2 – 3.4.

Chapter 4: Data Analysis and Interpretation

4.1 Introduction

Our initial project goal was to collect large amounts of data to train a neural network classifier. We scheduled a date to meet with many students participating in a biosignals lab in order to collect EEG signals from a diverse group of people. Due to COVID-19, we were unable to meet and were forced to continue to reach the end goal of our project using the limited data we had gathered so far. We also chose to focus on 2 channels of our recorded data, corresponding to electrodes C3 and C4. The experiments we had conducted so far were focused around mu rhythms which come from the motor cortex. The C3 and C4 electrodes are on the left and right hemispheres of the brain, respectively. The middle area on top of the head, Cz, is the electrode we used as our ground.

4.2 Experiments

After much trial and error with various tasks, the two tasks we settled on were to have the subject relaxed and to imagine performing a motor task. It was important to have both tasks involve no motor movement to keep within the design constraints of our project goals. The relaxed task instructs the subject to sit still in a chair with their eyes open and not to move. Keeping the eyes open is a very important part of the task. Closing the eyes can vastly alter brain waves, making our project extremely easy. Although, closing the eyes to stop would be extremely impractical as a control method for an arm or a wheelchair.

Therefore our relaxed task had to require the subject to keep their eyes open. The subject was not instructed on what to think about. They could freely focus or defocus on whatever they wanted as long as they weren't performing the second task.

The second task instructed the subject to focus on imagining performing a motor task. Like the previous task, it was important that the patient have their eyes open. This task instructed the patient to imagine moving their arm up and down in order to desynchronize the mu rhythms. These rhythms are synchronized when someone is at rest, and not moving, but should desynchronize when moving and, hopefully, when imagining to move.

4.3 Data Analysis

4.3.1 Fourier Transform

The purpose of a fourier transform is to analyze waveforms by breaking it down into its constituent frequencies (frequency analysis) and associated magnitude or power at each frequency component e.g. Power Spectral Density (PSD). Short-time Fourier Transform (STFT) is a time-frequency analysis method that takes the Fourier transform of sequential windowed segments, providing information on frequency makeups of the signal at each time window. In terms of this project, STFT is predicted to be the best tool for differentiating brain waves because of its 3-D graphical model which allows the three variables of interest (power, time, and frequency) to be represented together. On the other hand, the FFT provides information on the range of frequencies in the signal averaged over the entire time period. By transforming the data into frequencies, the data could be better classified with a machine learning model.

4.4 Feasibility Experiment

In order to test the feasibility of our project, we designed an experiment to see if our tasks and chosen brain areas of focus were on the right track. The end goal of the experiment relies on previous literature research that mu rhythms desynchronize when imagining to perform a task. Mu rhythms are known to desynchronize when performing a motor task, but imagining a task is difficult and we were not sure if a patient could focus enough for us to cause that desynchronization.

To test our tasks, we had the subject perform a third task to act as a baseline for what we expect to see from the imagining task. All three tasks were to be performed for a minute. This third task was to actually perform the motor task. This task goes against our project restriction, but it was an important step in our research to determine if the EEG waves from imagining a task are similar enough to the EEG waves of actually performing a motor task. Thus, if the data looks similar, our project is deemed feasible and we could continue with the tasks we designed. No future patients would be required to perform this task and we would not be using this data to train our model.

After recording data using our aforementioned data gathering procedure, we ran the data through three separate post-processing techniques. The first two that were used were Fast-Fourier Transform (FFT) and Power Spectral Density (PSD). Both of these processing techniques convert the graphs from time and amplitude axis to frequency and amplitude. The amplitude of the FFT spectrum is measured in energy while the amplitude of the PSD graph is measured in power. The conversion from the time domain to the frequency domain allows us to specifically identify the region of focus: the mu rhythms. The third technique, STFT, is a graphical representation of power, time, and frequency, allowing one to observe the power of the wave over the time and frequency domain.

4.5 Results

Our prediction and test for feasibility required the imagining of the motor task graph to match the graph of performing the task. When comparing the respective post-processing graphs together, the PSD graphs supported our mu rhythm desynchronization theory. The peak in **Figure 4.4** at roughly 12 Hz, which falls in the mu rhythm range of 7.5-12.5 Hz, represents a high power correlating to a synchronization of rhythms at 12 Hz. This peak is noticeably absent when the subject is waving their hand (**Figure 4.5**) and imagining the motion (**Figure 4.6**); instead, there are a bunch of smaller peaks with similar amplitudes.

However, the FFT and STFT graphs may be harder cases to sell when comparing their similarities. The FFT spectrum of the subject waving **Figure 4.2** has noticeably less peaks than the "at rest" (**Figure 4.1**) and the "imagining" spectra (**Figure 4.3**). Based on the FFT data, it may be concluded that the brain exhibits electrical activity when it is imagining a motion similarly to when it is at rest, which is the undesired conclusion. The STFT graphs differ more with each other than the FFT graphs. When comparing the bands of power/frequency that are greater than -60 dB/ Hz, the "at rest" graph (**Figure 4.7**) exhibits a narrower band than the other two graphs with a range from 8 to 13 Hz. The "waving hand" graph (**Figure 4.8**) exhibits the widest band that ranges from 8 to 22 Hz. The "imagining waving hand" graph (**Figure 4.9**) has a wider band than **Figure 4.7** but not as wide as **Figure 4.8.**

Based on our results, we concluded that our tasks were substantial enough to continue onto the next phase of our project. However, much more data is needed to confirm the similarities and differences between each of these graphs.
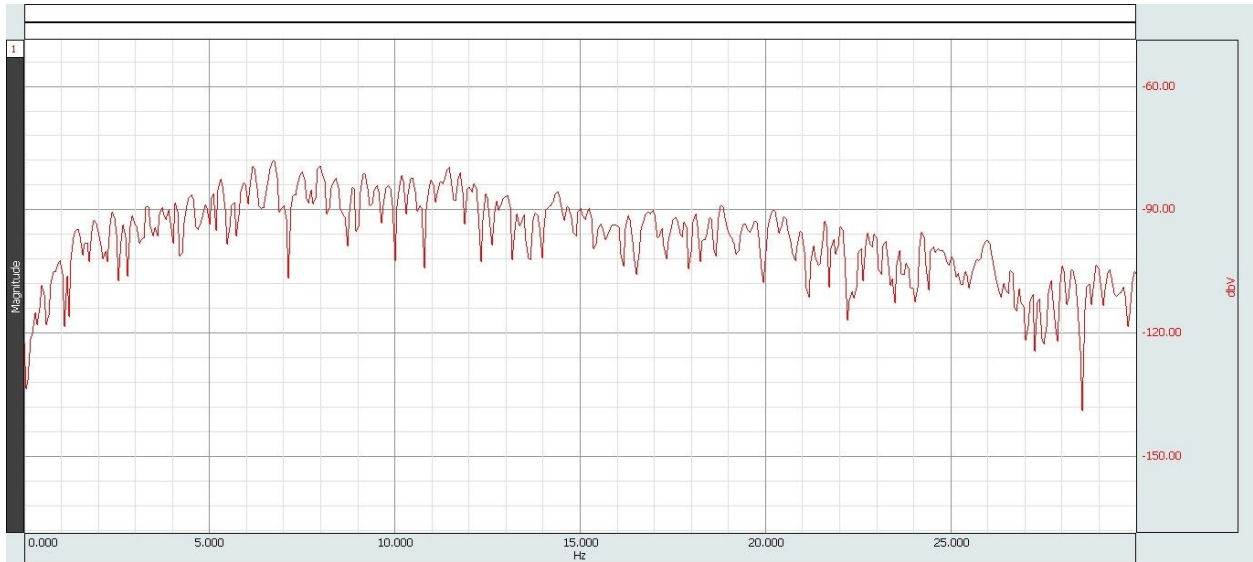
Figure 4.1: FFT of Subject at Rest. FFT of EEG data at electrode C3 (left motor cortex) when the subject is relaxed with their eyes open. The frequency range shown is from 0 to 30 Hz.

Figure 4.2: FFT of Waving Right Hand. FFT of EEG data at electrode C3 (left motor cortex) when the subject is waving their hand. The frequency range shown is from 0 to 30 Hz.

Figure 4.3: FFT of Imagining Waving Right Hand FFT of EEG data at electrode C3 (left motor cortex) when the subject is imagining waving their hand. The frequency range shown is from 0 to 30 Hz.

Figure 4.4 PSD of Subject at Rest. PSD of EEG data at electrode C3 (left motor cortex) when the subject is relaxed with their eyes open. The frequency range shown is from 0 to 30 Hz.
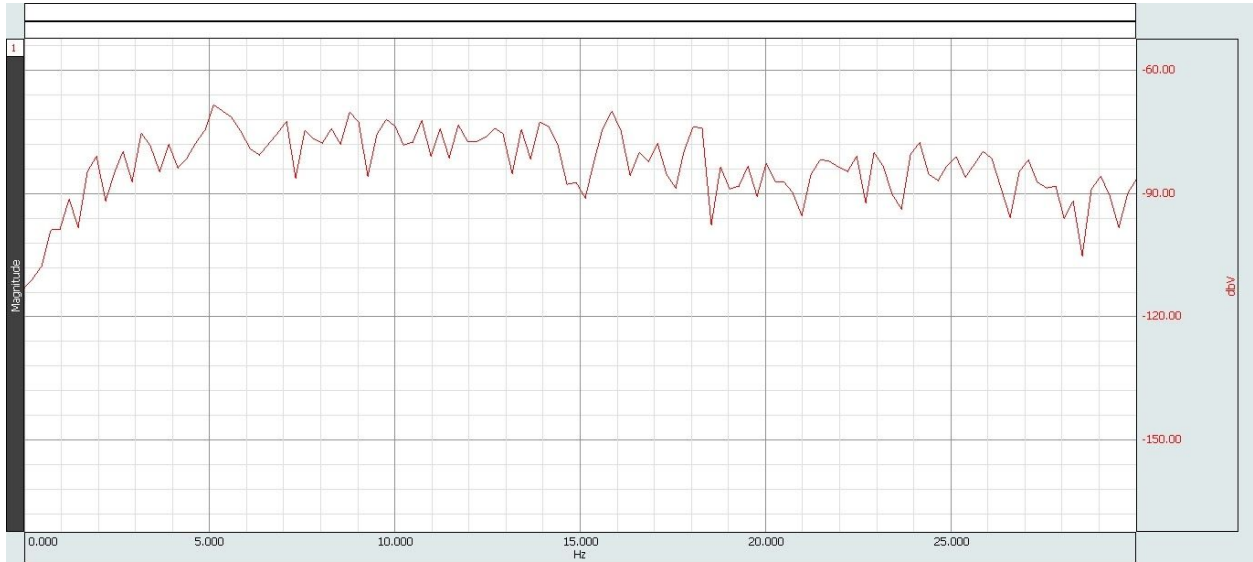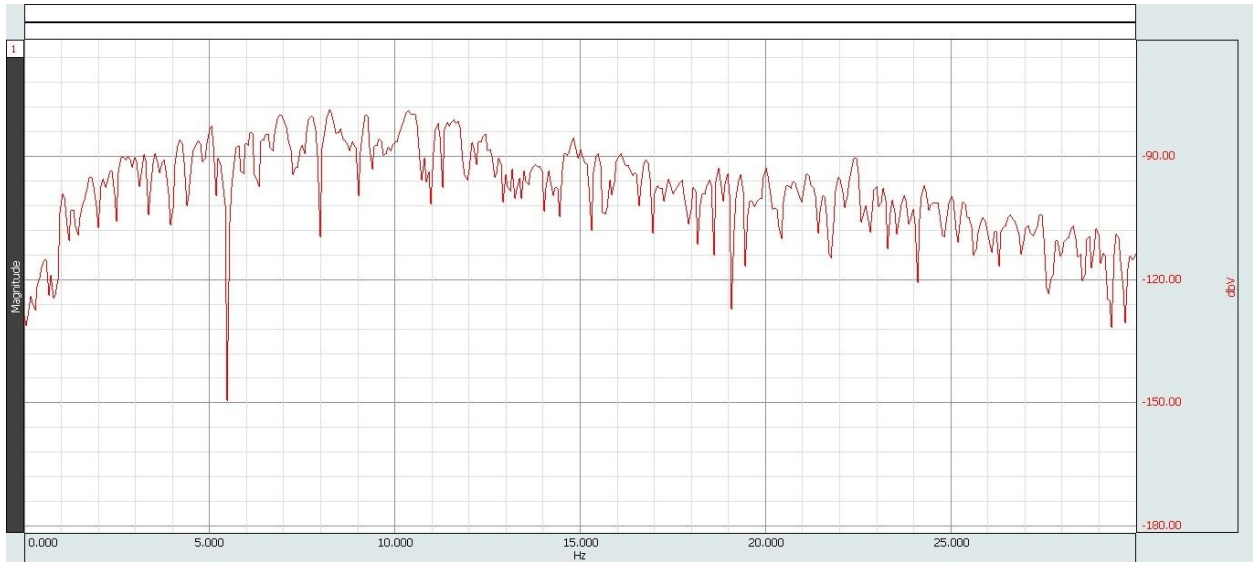
Figure 4.5 PSD of Waving Right Hand. PSD of EEG data at electrode C3 (left motor cortex) when the subject is waving their hand. The frequency range shown is from 0 to 30 Hz.

Figure 4.6 PSD of Imagining Waving Right Hand. PSD of EEG data at electrode C3 (left motor cortex) when the subject is imagining waving their hand. The frequency range shown is from 0 to 30 Hz.
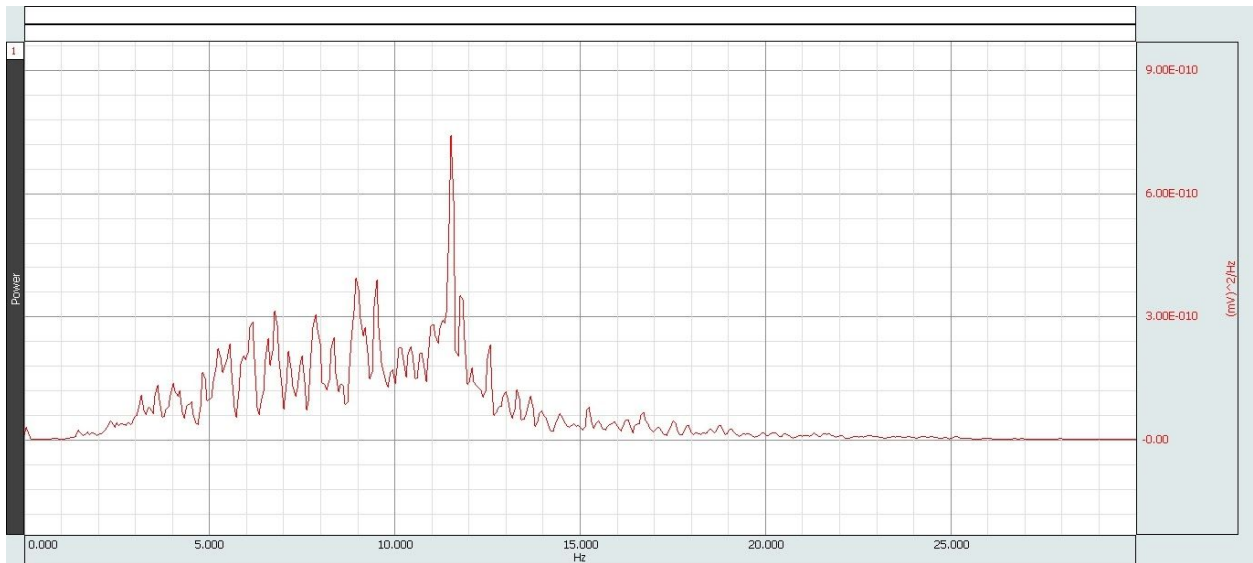
Figure 4.7 STFT of Subject at Rest. Spectrogram of EEG data at electrode C3 (left motor cortex) when the subject is relaxed with their eyes open.
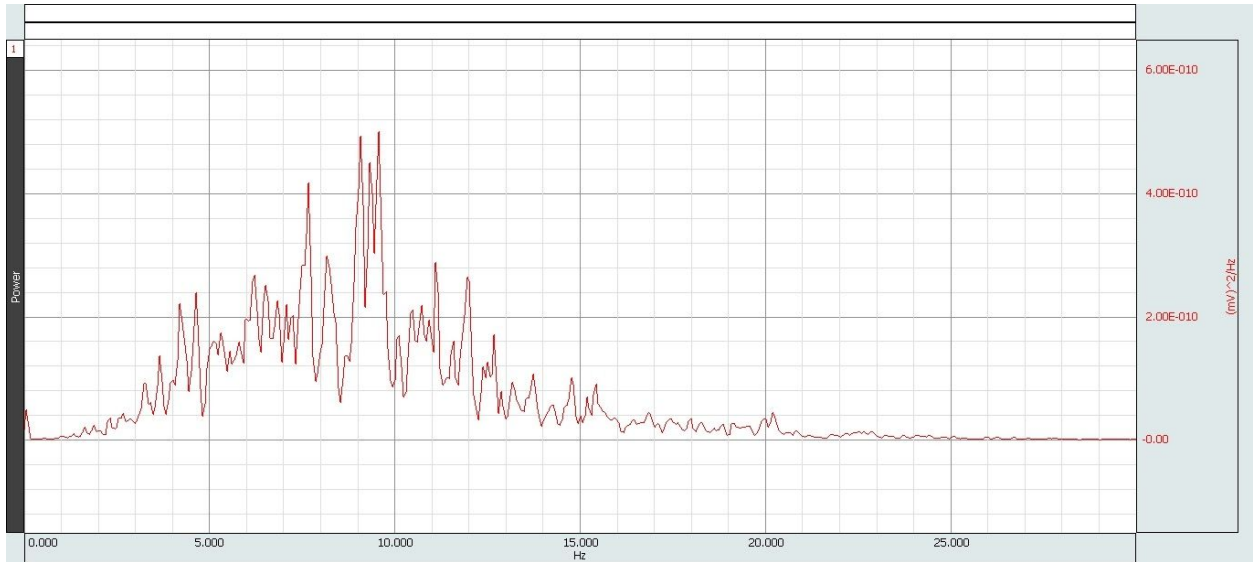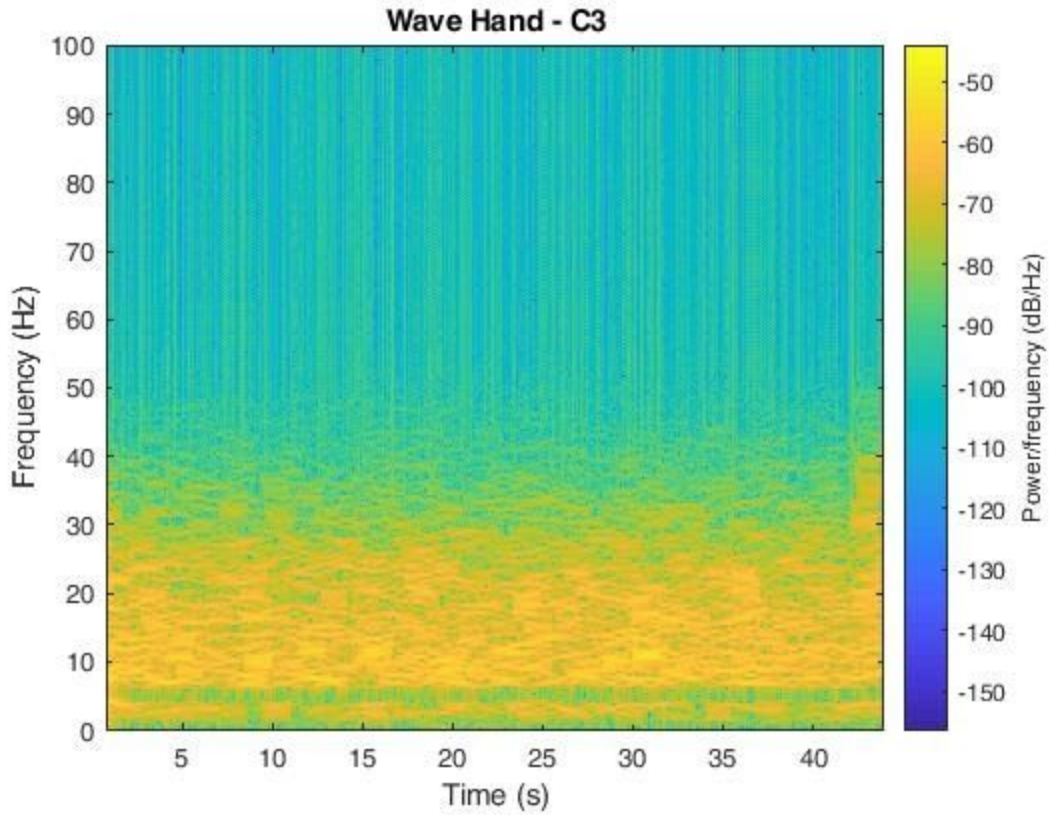
Figure 4.8 STFT of Waving Right Hand. Spectrogram of EEG data at electrode C3 (left motor cortex) when the subject is waving their hand.
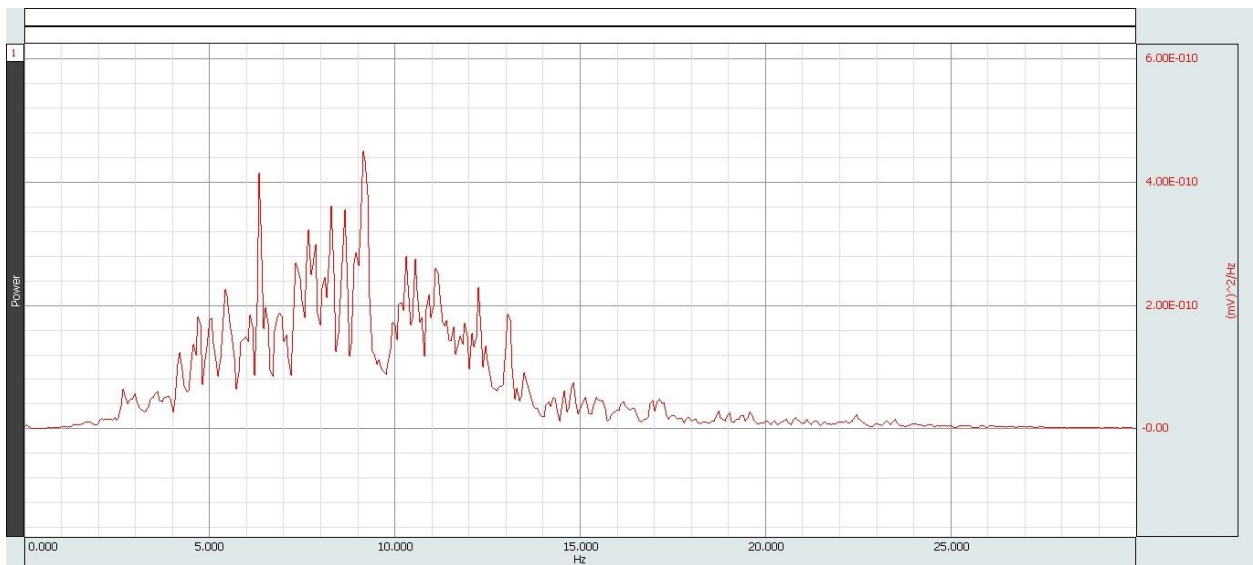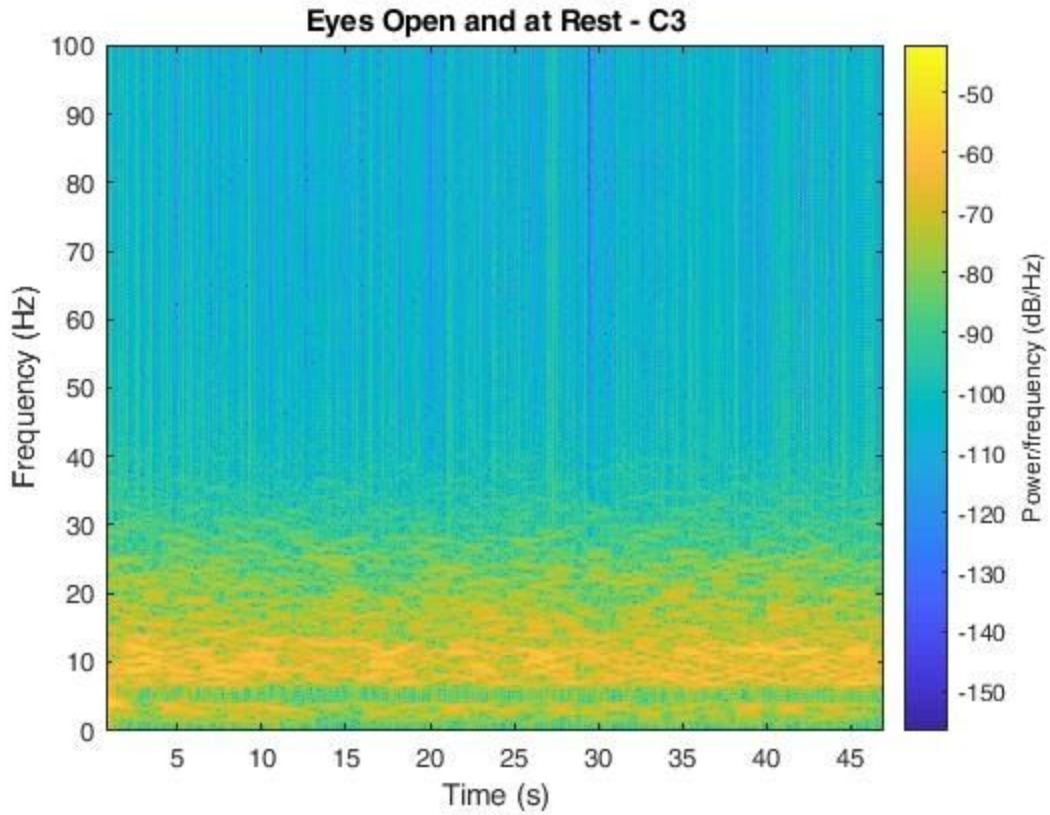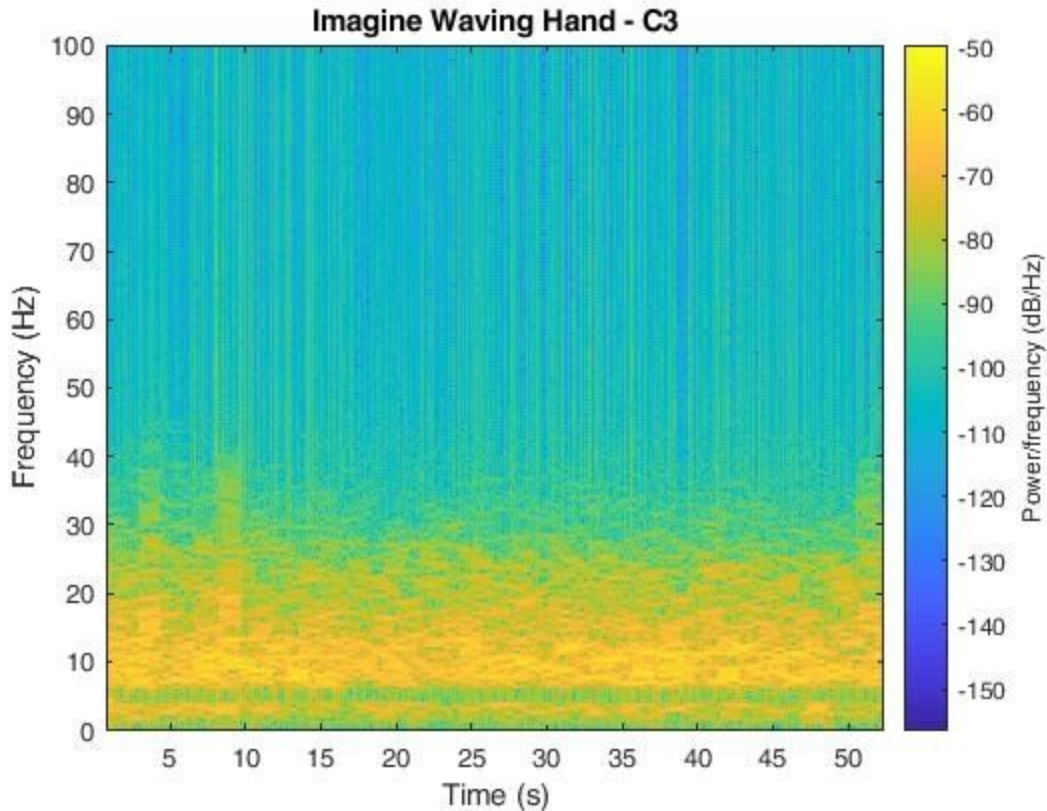
Figure 4.9 STFT of Imagining Waving Right Hand. Spectrogram of EEG data at electrode C3 (left motor cortex) when the subject is imagining waving their hand.

4.6 Future Steps

The data recorded was a minute long, which makes it much easier to determine the desynchronization of mu rhythms. When analyzing data real time, data should not be sent at a latency of one minute. This would be impractical. Now that the tasks were proven to work, the next step is to perform the PSD post-processing techniques using data at a much lower resolution. Our initial goal was to have a latency of 1 second, and therefore sending data with a resolution of 1 second. In order to set a more realistic goal, we decided to first scale down the minute resolution to 4 seconds.

Chapter 5: Machine Learning Implementation

5.1 Introduction

The process of machine learning consists of a system of algorithms that uses inference and, subsequently, pattern recognition, to perform a specific task without instructions. There are many forms of machine learning that are used for different

purposes, but the type of model that is of the most interest is Support Vector Machine (SVM), a type of machine learning model similar to that of logistic regression. In a dataset that can be graphed in a 2D or 3D space, SVM creates a hyperplane, a plane that contains the maximum distance between two data points, that separates the data in two distinct classes [8].

Using SVM to classify brain signals has been done in 2013 in a study conducted by Bayram et al. In this experiment, Bayram and his team attempted to classify brain waves by two categories:

1. The patient is planning to do a task.
2. The patient is at rest.

To do this, Bayram instructed the patient to rest for five minutes at rest with their eyes shut. Then, a beep sound of 60 dB would be played which signaled the subject to plan movement of the right thumb for five seconds. After filtering the data, feature selection methods were used to further reduce the size of the data and improve classification accuracy. With this experimental setup, they obtained a selection accuracy of about 72% [16]. In terms of this project, this accuracy is concerning due to its unreliability (> 95%). Since our goal requires classification between two different classes, it would be worthwhile to find a way to increase the accuracy of this experiment.

5.2 Datasets

Our initial plan when we started this project was to train a convolutional neural network (CNN) using a large, diverse dataset. As aforementioned, we weren't able to record a large dataset due to social distancing and shelter in place. With unlucky timing, shelter in place was issued a few days before the scheduled date, and we were left with a significantly smaller, and less diverse dataset. The plan was to convert the signals into spectrograms and use the CNN to extract the features to classify the testing set of spectrograms into the focused and unfocused states. Because of our limited dataset, we decided to lower the resolution of the samples significantly and extract the features ourselves. We then ran these features into a support vector machine (SVM) to see if the algorithm could correctly determine the state with a lower resolution.

5.3 Machine Learning Libraries

The SVM model was created from scratch using MATLAB and Python. The logic of the model is as follows:

1. The EEG data is quantified in such a way that the machine learning model can classify it. To do this, we decided to take the root-mean-square (rms) and the average frequencies of an entire sample to represent one data point to be classified by the model. The rms was collected because it is proportional to the power of the entire wave [17]. There were a total of 100 data points. We planned to take the spectrogram from each sample and used it for classification because classifying rms

values did not feel enough. However, due to time constraints, only the spectrograms were obtained.

2. The data was classified without assistance from a model. Each data point is assigned a zero if the data came from at rest or a one if the data came from task imagination.
3. Using the extracted data, we created variables "X_train", "y_train", "X_test" and "y_test" to randomly create similar data points.
4. The best features were taken out of the training and testing data sets
5. Using the best features, a scatterplot would be plotted and a hyperplane would be created.

Refer to **Appendix D** for the code of the previously mentioned logic.

5.4 Results

Based on the code in Appendix D and our collected data, our model graphically represented in **Figure 5.1** was able to accurately classify the rms value to its brain state 60% of the time. Like the Bayram experiment, this accuracy is concerning because it is akin to the model flipping a coin to decide if the rms value matches the brain state. The low accuracy may be attributed to the data set that was fed into the model. Table 5.1 shows a sample of rms values that the model compared with each other:

RMS value	Average Frequency (Hz)	State
0.0021	9.2175	Rest/ Eyes Open
0.0017	9.9393	Rest/ Eyes Open
0.0019	10.3548	Rest/ Eyes Open
0.0016	9.7968	Rest/ Eyes Open
0.0021	9.7740	Rest/ Eyes Open
0.0018	10.0360	Imagine Wave Hand
0.0020	9.5246	Imagine Wave Hand
0.0027	9.8675	Imagine Wave Hand
0.0022	9.2490	Imagine Wave Hand
0.0019	9.0418	Imagine Wave Hand

Table 5.1 RMS Values vs. Average Frequencies. These values were extracted from four second wave segments from one minute recordings of the Eyes-Open/Rest task and the Imagining Waving Hand task.

From glancing at the RMS and average frequencies, there is no observable difference between Rest/Eyes Open and Imagine Waving Hand because the sets of their respective RMS and frequencies overlap greatly. Because of this phenomenon, a separation line could not be easily drawn to truly differentiate the two, leading to the "coin-flip" accuracy.
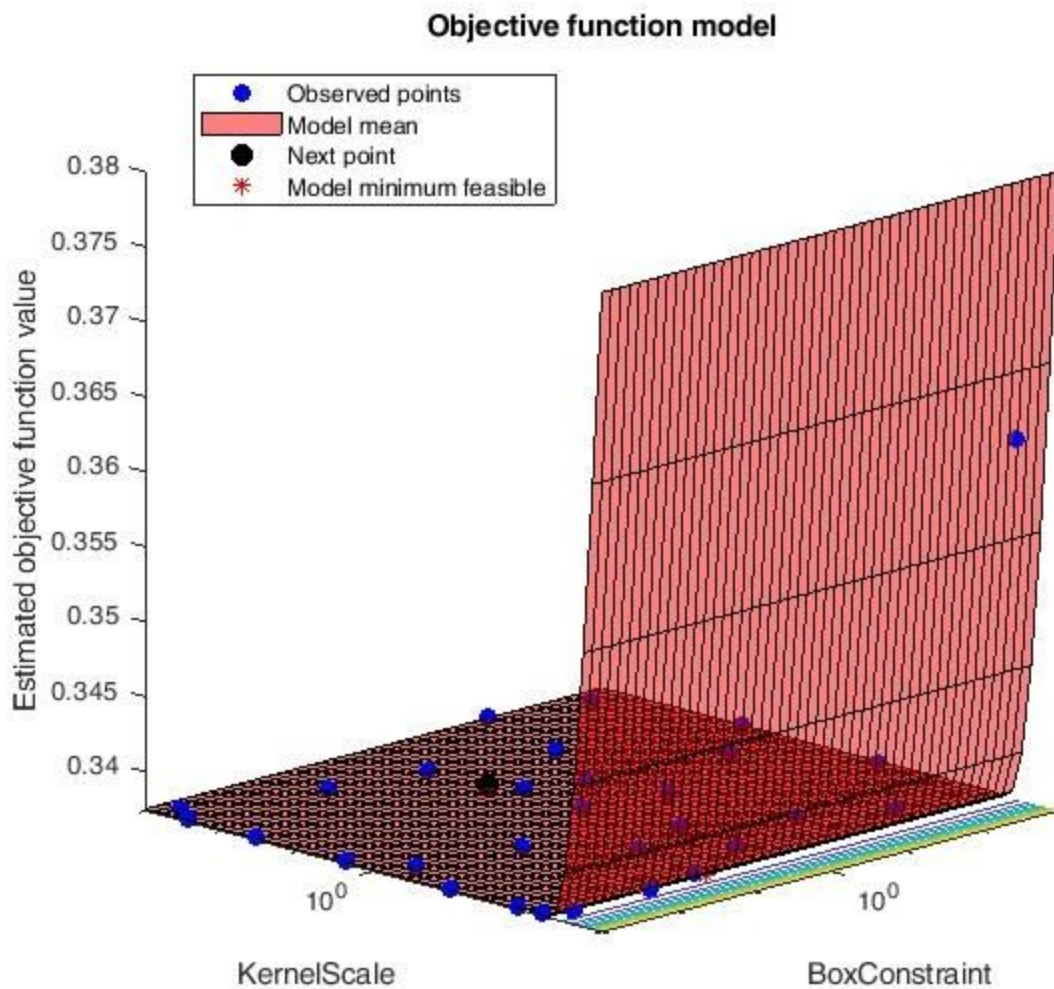

Figure 5.1 Radial Basis Function (RBF) Model. The model represents the rms values and its respective brain state category using kernels. The kernels are evaluated by their estimated objective function value with which the model uses to create the model mean.

5.5 Future Steps

To achieve greater accuracy in the SVM, it is recommended to classify additional features along with the rms and the average frequencies of the wave. Spectrograms of the wave sample delineates the power of the wave at certain frequencies over time by a usually distinguishable band. If one wishes to incorporate spectrograms into their SVM model, they would likely need to convert the graph into a grayscale image where the distinguishable band is white. They would need to quantify these images and put all of these images into a matrix that would be run into the model. Along with additional features, we were going to analyze different electrodes in the 10/20 map and compare their features with each other for any observable differences.

There may have been some equipment issues when collecting the data that may have affected overall results. Because the budget did not allow for an impedance checker, we were unable to check for impedance throughout all of the electrodes, arguably a crucial step in brainwave data collection. By ignoring possible impedance, our collected data may have been weaker than what it actually should be. However, electric signals from the brain are generally weak. The electro-cap provided was too small causing the placement of the electrodes to be sub-optimal for our experiment.

Practically, this project would not have worked because there was no variety in the data, and, therefore, would only work with a select amount of people. The data was limited to one subject and a few recordings of the tasks that were designed. To achieve better variety in data, it is recommended to have a sample size of at least 30 different subjects with varying physical ability. Because tasks can be focused on for a limited amount of time before being distracted, it is recommended that the tasks either be more engaging or, for imagination tasks, be executed for at most ten seconds. With these tasks in mind, it would also be useful to somehow measure the attentiveness of the subject as they may have an effect on the frequency bands.

Chapter 6: Robotic Implementation (Proof of Concept)
6.1 Introduction

Our initial plan to implement our project into a robotic system was cut short to budget constraints as well as the COVID-19 pandemic. We had planned to use our trained neural network in order to read EEG signals in real time and send those commands to a remote control car reprogrammed using an Arduino. Unfortunately, our budget was not approved to cover the Network Data Transfer (NDT) license so we weren't able to test our project with a reprogrammable car.

6.2 Suggested Materials and Methods

Suggested materials are the materials and software that were not approved in our requested budget.

1. Network Data Transfer (NDT) license
2. Impedance checker
3. Programmable remote control car
4. Arduino
5. Car parts and wires

The NDT will be used to transfer data to a remote control car real time to demonstrate the machine learning. The impedance checker will greatly reduce the noise of the signals to allow for much more accurate post-processing and neural network training.

6.3 Hypothetical Results

The reported number we were able to achieve with our limited dataset, 60%, represents our test results. If we were able to purchase the NDT and the car, we would be able to feed in EEG data real-time with an accuracy of 60%. The car would be able to move 60% of the desired time with the current model that we trained. This number would potentially greatly improve with the future steps we mentioned throughout our thesis. Due to COVID-19 we can't implement our test results further, but our project could be taken up by future groups to integrate into a system to demonstrate the network's accuracy. Hopefully any hypothetical future group would be able to increase the accuracy significantly with a larger dataset using our methodology, algorithm, and tasks and be able to send the data in real-time using the NDT.

Chapter 7: Engineering Standards

The ethical, social, economic, and manufacturability engineering standards have impacted this device's design the most.

7.1 Ethical

This method of brain-wave extraction is non-invasive, meaning that no part of the device was inserted into the subject's body. Because of the electrodes' indirect contact with the brain, the recordings are limited to the activities on the cortex. Despite its proximity, non-invasive EEG provides enough information about the brain to measure differences in its various states and subsequently diagnose diseases.

To obtain more accurate data with higher amplitudes, one may consider invasive procedures to get the electrodes directly onto the cortex and possibly in the innermost layers of the brain. In terms of our project, however, while invasive EEG leads to more accurate results, it is not the most ideal. Invasive EEG procedures will require expensive

brain surgery to attach the electrodes directly onto the brain which also runs risks of infection and possible biocompatibility issues. By sacrificing accuracy in favor of a more humane way to collect brain data, this product would be more marketable to the public, and subjects would be more willing to try the device.

7.2 Social

This model is intended to be used by everyone with varying physical and mental ability. In lieu of physical tasks, mental tasks were planned and executed to target the entire physically abled population. By eliminating physical requirements, this device will increase confidence and independence for motor disabled patients and increase the possibility to "mind control" unintended things.

The scope of this project targeted the physically abled and disabled, mentally abled population. For this product to truly be universal and inclusive, additional restrictions must be placed on the tasks to cater to the mentally abled and disabled population.

7.3 Economic

Our project focuses on creating a control method that could be available to everyone, at the price of not being affordable for everyone. There are already cheap alternatives that exist for motorly disabled, such as wheelchairs, but there are not many options for the severely motorly impaired. If implemented, this technology would offer a much needed option but it would probably be very expensive in terms of equipment.

In addition to expensive equipment, the patient's expenses may include services. Potential services could be physical therapy, training, as well as refining the neural network and data processing to fit the specific patient. These services could further drive the cost upwards.

Although this technology would be extremely expensive, if adapted and improved it could also save money for many people in the far future. Patients that require the aid of caretakers could gain independence through brain-controlled control methods. Instead of paying to be taken care of, patients could pay to take care of themselves.

7.4 Manufacturability

The manufacturability of this method will vary depending on the device that is wished to be controlled. Our project is the foundation of a control method that converts brain waves to commands that is not specifically tied to a device. A manufacturer who wishes to implement this control method must alter this control method to fit the needs of their respective devices.

Chapter 8: Conclusions

The 60% accuracy in our model implies that 5 second resolution using RMS values is not enough for classification, much less robotic control. If we were to increase the sampling time and, therefore, resolution, we could have more distinct features between graphs suitable for classification. In our PSD experiment, because of the relative similarity between the wave hand and imagine wave hand graphs, there is cause to believe that imagining movement desynchronizes mu rhythms the same way that movement does. It was also supported that the brain region of focus, the central lobe, was ideal for collecting mu rhythms at approximately 12 Hz. The 60 second resolution PSD was enough to easily distinguish the focused and unfocused states, but we didn't have a big enough sample size.

In the future, these tasks and processing methods could be used for a much larger and diverse dataset. The larger dataset would enable a convolutional neural network (CNN) to be used instead of a support vector machine (SVM). We also recommend testing subjects with varying physical or mental ability to better fit the overall goal of the project. Instead of extracting features, such as the RMS values, to run through the SVM, we would run two or three dimensional graphs into the CNN to extract features. We recommend sending either PSD or spectrogram graphs into a CNN. The results of our experiments supported that the frequency and power domains were crucial to reading the EEG signals. We also recommend using the same tasks as a brain area of focus for the electrode placement to properly assess the mu rhythm desynchronization.

Appendices

Appendix A: Eyes Open and at Rest

```
%% EyesOpen_Rest2_C3
close all
clear all
clc

[soundx,fs] = audioread('EyesOpen_Rest2_C3.wav'); %% load the sound from wave to ascii
format

%downsampling (or apply low pass instead of down sample and bandpass)
soundx=downsample(soundx,10);

fs=(fs/10); %frequency rate (Hz)
fn = fs/2; %nyquist frequency (Hz)
N = length(soundx);
T = (1:N)/fs;
sze = length(soundx);
x = (1:sze)/fs;    %% scaling the x axis to the actual scale


figure;                            %% plot the sound wave
plot(x, soundx)
axis([0 50 -.02 .02])
xlabel('Time (s)');
ylabel('Amplitude (V)');
title('Rest Original EEG Signal');

d= designfilt('bandstopiir', 'FilterOrder',10,...
   'HalfPowerFrequency1',8/fs,'HalfPowerFrequency2',13/fs,...
   'DesignMethod','butter');
fvtool(d,'fs',fs)

buttLoop=filtfilt(d,soundx);

plot(x,buttLoop)
axis([0 50 -.02 .02])
ylabel('Amplitude (V)')
xlabel('Time (s)');
```

```
title('Rest Filtered EEG Signal');
grid

% Original vs. Filtered
plot(x,soundx,x,buttLoop)
axis([0 50 -.01 .01])
ylabel('Amplitude (V)')
xlabel('Time (s)');
title('Eyes Open and at Rest');
legend('Unfiltered','Filtered');
grid

%%%%%%%%% STFT analysis/Spectrogram plot
%T=1024/fs;    %%% sample length = N * dt (sampling interval) = N/fs
df=1./T;       % frequency resolution for the FFT analysis of each segment
freq=0:df:fs/2;      % define frequency axis

figure;
%spectrogram(soundx,1024,1000,freq,fs,'yaxis')
spectrogram(buttLoop,kaiser(301,0.5),300,1000,fs,'yaxis'); % increase NOVERLAP, 50 is too
small --> 300, zoom in to a few seconds, freq resolution p bad, for original eeg,
title('Eyes Open and at Rest - C3');
```

Appendix B: Imagining Waving Hand

```
%% ImagineWaveHand_C3
close all
clear all
clc

[soundx1,fs] = audioread('ImagineWaveHand_C3.wav'); %% load the sound from wave to
ascii format

%downsampling
soundx1=downsample(soundx1,10);
fs=(fs/10); %frequency rate (Hz)
fn = fs/2; %nyquist frequency (Hz)
N = length(soundx1);
T = (1:N)/fs;
sze = length(soundx1);
x = (1:sze)/fs;    %% scaling the x axis to the actual scale
```

```matlab
figure;                          %% plot the sound wave
plot(x, soundx1)
axis([0 50 -.02 .02])
xlabel('Time (s)');
ylabel('Amplitude (V)');
title('Imagine Original EEG Signal');

d= designfilt('bandstopiir', 'FilterOrder',10,...
    'HalfPowerFrequency1',8/fs,'HalfPowerFrequency2',13/fs,...
    'DesignMethod','butter');
fvtool(d,'fs',fs)

buttLoop1=filtfilt(d,soundx1);

plot(x,buttLoop1)
axis([0 50 -.02 .02])
ylabel('Amplitude (V)')
xlabel('Time (s)');
title('Imagine Filtered EEG Signal');
grid

% Original vs. Filtered
plot(x,soundx1,x,buttLoop1)
axis([0 50 -.01 .01])
ylabel('Amplitude (V)')
xlabel('Time (s)');
title('Imagine Waving Hand');
legend('Unfiltered','Filtered');
grid

%%%%%%%%%% STFT analysis/Spectrogram plot
%T=1024/fs;     %%% sample length = N * dt (sampling interval) = N/fs
df=1./T;       % frequency resolution for the FFT analysis of each segment
freq=0:df:fs/2;      % define frequency axis

figure;
%spectrogram(soundx,1024,1000,freq,fs,'yaxis')
spectrogram(buttLoop1,kaiser(301,0.5),300,1000,fs,'yaxis');
title('Imagine Waving Hand - C3');
```

Appendix C: Waving Hand

```
%% WaveHand_C3
close all
clear all
clc

[soundx2,fs] = audioread('WaveHand_C3.wav'); %% load the sound from wave to ascii
format

%downsampling
soundx2=downsample(soundx2,10);
fs=(fs/10); %frequency rate (Hz)
fn = fs/2; %nyquist frequency (Hz)
N = length(soundx2);
T = (1:N)/fs;
sze = length(soundx2);
x = (1:sze)/fs;    %% scaling the x axis to the actual scale

figure;                         %% plot the sound wave
plot(x, soundx2)
axis([0 50 -.02 .02])
xlabel('Time (s)');
ylabel('Amplitude (V)');
title('Wave Hand Original EEG Signal');

d= designfilt('bandstopiir', 'FilterOrder',10,...
   'HalfPowerFrequency1',8/fs,'HalfPowerFrequency2',13/fs,...
   'DesignMethod','butter');
fvtool(d,'fs',fs)

buttLoop2=filtfilt(d,soundx2);

plot(x,buttLoop2)
axis([0 50 -.02 .02])
ylabel('Amplitude (V)')
xlabel('Time (s)');
title('Wave Hand Filtered EEG Signal');
grid


% Original vs. Filtered
```

```
plot(x,soundx2,x,buttLoop2)
axis([0 50 -.01 .01])
ylabel('Amplitude (V)')
xlabel('Time (s)');
title('Waving Hand - C3');
legend('Unfiltered','Filtered');
grid
% frequency spectrum --> entire saMPLE , or stft with time spectrum --> each segment,
time info
%%%%%%%%%% STFT analysis/Spectrogram plot
%T=1024/fs;      %%% sample length = N * dt (sampling interval) = N/fs
df=1./T;      % frequency resolution for the FFT analysis of each segment
freq=0:df:fs/2;      % define frequency axis

figure;
%spectrogram(soundx,1024,1000,freq,fs,'yaxis')
spectrogram(buttLoop2,kaiser(301,0.5),300,1000,fs,'yaxis');
title('Wave Hand - C3');
```

Appendix D: SVM

Parts of this code were pulled from "MATLAB SVM tutorial (fitcsvm) by Exploring the Meaning of Math [18] and adapted to fit our dataset. Refer to **Appendix E** for the user-created functions.

```
%% Value Extraction

y1= eegrms('EyesOpen_C3_1.wav');
y2= eegrms('EyesOpen_C3_2.wav');
y3= eegrms('EyesOpen_C3_3.wav');
y4= eegrms('EyesOpen_C3_4.wav');
y5= eegrms('EyesOpen_C3_5.wav');
y6= eegrms('EyesOpen_C3_6.wav');
y7= eegrms('EyesOpen_C3_7.wav');
y8= eegrms('EyesOpen_C3_8.wav');
y9= eegrms('EyesOpen_C3_9.wav');
y10= eegrms('EyesOpen_C3_10.wav');

fs1 = FreqExtract('EyesOpen_C3_1.wav');
fs2 = FreqExtract('EyesOpen_C3_2.wav');
fs3 = FreqExtract('EyesOpen_C3_3.wav');
```

```matlab
fs4 = FreqExtract('EyesOpen_C3_4.wav');
fs5 = FreqExtract('EyesOpen_C3_5.wav');
fs6 = FreqExtract('EyesOpen_C3_6.wav');
fs7 = FreqExtract('EyesOpen_C3_7.wav');
fs8 = FreqExtract('EyesOpen_C3_8.wav');
fs9 = FreqExtract('EyesOpen_C3_9.wav');
fs10 = FreqExtract('EyesOpen_C3_10.wav');

eyesopenrms= [y1, y2, y3, y4, y5, y6, y7, y8, y9 , y10]

z1= eegrms('Imagine_C3_1.wav');
z2= eegrms('Imagine_C3_2.wav');
z3= eegrms('Imagine_C3_3.wav');
z4= eegrms('Imagine_C3_4.wav');
z5= eegrms('Imagine_C3_5.wav');
z6= eegrms('Imagine_C3_6.wav');
z7= eegrms('Imagine_C3_7.wav');
z8= eegrms('Imagine_C3_8.wav');
z9= eegrms('Imagine_C3_9.wav');
z10= eegrms('Imagine_C3_10.wav');

fs11 = FreqExtract('Imagine_C3_1.wav');
fs12 = FreqExtract('Imagine_C3_2.wav');
fs13 = FreqExtract('Imagine_C3_3.wav');
fs14 = FreqExtract('Imagine_C3_4.wav');
fs15 = FreqExtract('Imagine_C3_5.wav');
fs16 = FreqExtract('Imagine_C3_6.wav');
fs17 = FreqExtract('Imagine_C3_7.wav');
fs18 = FreqExtract('Imagine_C3_8.wav');
fs19 = FreqExtract('Imagine_C3_9.wav');
fs20 = FreqExtract('Imagine_C3_10.wav');

imaginerms= [z1, z2, z3, z4, z5, z6, z7, z8, z9, z10]

%wavehand data -- taken out 5/4/20
% zq1= eegrms('WaveHand_C3_1.wav');
% zq2= eegrms('WaveHand_C3_2.wav');
% zq3= eegrms('WaveHand_C3_3.wav');
% zq4= eegrms('WaveHand_C3_4.wav');
% zq5= eegrms('WaveHand_C3_5.wav');
% zq6= eegrms('WaveHand_C3_6.wav');
% zq7= eegrms('WaveHand_C3_7.wav');
% zq8= eegrms('WaveHand_C3_8.wav');
```

```matlab
% zq9= eegrms('WaveHand_C3_9.wav');
% zq10= eegrms('WaveHand_C3_10.wav');
%
% fs21 = FreqExtract('WaveHand_C3_1.wav');
% fs22 = FreqExtract('WaveHand_C3_2.wav');
% fs23 = FreqExtract('WaveHand_C3_3.wav');
% fs24 = FreqExtract('WaveHand_C3_4.wav');
% fs25 = FreqExtract('WaveHand_C3_5.wav');
% fs26 = FreqExtract('WaveHand_C3_6.wav');
% fs27 = FreqExtract('WaveHand_C3_7.wav');
% fs28 = FreqExtract('WaveHand_C3_8.wav');
% fs29 = FreqExtract('WaveHand_C3_9.wav');
% fs30 = FreqExtract('WaveHand_C3_10.wav');
%
% wavehandrms= [zq1, zq2, zq3, zq4, zq5, zq6, zq7, zq8, zq9, zq10]; %theres a noticeable
difference in rms values comparing eyes open and wave hand. Imagine and eyes open are
relatively similar however. Maybe from phantom movement it would be okay?

A= [eyesopenrms, imaginerms];
%A= [eyesopenrms, imaginerms, wavehandrms];
B= [fs1, fs2, fs3, fs4, fs5, fs6, fs7, fs8, fs9, fs10, fs11, fs12, fs13, fs14, fs15, fs16, fs17, fs18, fs19,
fs20];

%B= [fs1, fs2, fs3, fs4, fs5, fs6, fs7, fs8, fs9, fs10, fs11, fs12, fs13, fs14, fs15, fs16, fs17, fs18, fs19,
fs20, fs21, fs22, fs23, fs24, fs25, fs26, fs27, fs28, fs29, fs30];
Q= [A, A, A, A, A];
W= [B, B, B, B, B];

%% SVM
% States
aa= zeros(10,1) % 0 = rest
bb= ones(10,1) % 1 = imagine
%cc= 2*ones(10,1) % 2 = wave
dd = [aa; bb; aa; bb; aa; bb; aa; bb; aa; bb]
%dd = [aa; bb; cc; aa; bb; cc; aa; bb; cc; aa; bb; cc; aa; bb; cc]

% Binary Classification
X = [Q']% Q= rms values, W= frequencies
%X= [Q';W']
y = dd(1:100) % States: rest (0), imagine (1)

% 80:20
rand_num = randperm(size(X,1)) %randomly permutate integers
```

```
X_train = X(rand_num(1:round(0.8*length(rand_num))),:)
y_train = y(rand_num(1:round(0.8*length(rand_num))),:)

X_test = X(rand_num(round(0.8*length(rand_num))+1:end),:)
y_test = y(rand_num(round(0.8*length(rand_num))+1:end),:)

%% CV partition

c = cvpartition(y_train,'k',5);
% feature selection

opts = statset('display','iter');
classf =  @(train_data, train_labels, test_data, test_labels)...
sum(predict(fitcsvm(train_data, train_labels,'KernelFunction','rbf'),
test_data)~=test_labels)


[fs, history] = sequentialfs(classf, X_train, y_train,'cv',c,'options',opts,'nfeatures',1);
%% Best hyperparameter

X_train_w_best_feature = X_train(:,fs)

Md1 =
fitcsvm(X_train_w_best_feature,y_train,'KernelFunction','rbf','OptimizeHyperparameters'
,'auto',...
    'HyperparameterOptimizationOptions',struct('AcquisitionFunctionName',...
    'expected-improvement-plus','ShowPlots', true)); % Bayes' Optimization ??.
% predict(Md1, X_train_w_best_feature)

%% Final test with test set
X_test_w_best_feature = X_test(:,fs)
test_accuracy_for_iter = sum((predict(Md1,X_test_w_best_feature) ==
y_test))/length(y_test)*100
```

Appendix E: User-Created Functions

% EEGRMS

```
function y = eegrms(filename);

[soundx,fs] = audioread(filename); %% load the sound from wave to ascii format

%downsampling (or apply low pass instead of down sample and bandpass)
soundx=downsample(soundx,10);

fs=(fs/10); %frequency rate (Hz)
fn = fs/2; %nyquist frequency (Hz)
N = length(soundx);
T = (1:N)/fs;
sze = length(soundx);
x = (1:sze)/fs;    %% scaling the x axis to the actual scale
d= designfilt('bandstopiir', 'FilterOrder',10,...
   'HalfPowerFrequency1',8/fs,'HalfPowerFrequency2',13/fs,...
   'DesignMethod','butter');
fvtool(d,'fs',fs)

buttLoop=filtfilt(d,soundx);

y= rms(buttLoop)
end
```

%FREQEXTRACT

```
function freq = FreqExtract(folder)
[soundx,fs] = audioread(folder);

freq = meanfreq(soundx,fs);
end
```

References

1. Christopher & Dana Reeve Foundation. (2013). *Stats about paralysis*. Christopher & Dana Reeve Foundation.
2. National Spinal Cord Injury Statistical Center. (2019) *Facts and Figures at a Glance*. University of Alabama at Birmingham.
3. FlintRehab. (2019). *Physical Therapy for Paraplegia: Why It's Necessary for SCI Recovery*. FlintRehab.
4. Nas, K., Yazmalar, V.S., Aydin, A., & Ones, K. (2015). Rehabilitation of spinal cord injuries. *World Journal of Orthopedics*, 6(1), 8-16.
5. Blocka, K. (2017). *EEG(Electroencephalogram)*. Healthline.
6. Yan. Y (2019, January 15). *Lecture 3: Bioelectric Signals ECG/ EEG/ EMG*. Santa Clara University, Santa Clara, CA.
7. Hobson H.M. & Bishop D.V.M. (2017, March). *The interpretation of mu suppression as an index of mirror neuron activity: past, present and future*. Royal Society Open Science.
8. Gandhi, R. (2018). *Support Vector Machine – Introduction to Machine Learning Algorithms: SVM model from scratch*. Medium.
9. Yoshioka, M., Zhu, C., Imamura, K., Wang, F., Yu, H., Duan, F., & Yan, Y. (2014). Experimental design and signal selection for construction of a robot control system based on EEG signals. *Robotics and Biomimetics* 2014. 1-22.
10. Baranauskas, G. (2014). What limits the performance of current invasive brain machine interfaces?. *Frontiers in Systems Neuroscience*. 8(68).
11. Stach, T., Browarsa, N., & Kawala-Janik, A. (2018). Initial Study on Using Emotiv EPOC+ Neuroheadset as a Control Device for Picture Script-Based Communicators. *IFAC PapersOnLine*. 180-184.
12. Khare V., Santhosh J., Anand S., Bhatia M. (2011). Brain-computer interface based real time control of wheelchair using electroencephalogram. *International Journal of Soft Computing and Engineering (IJSCE)*. 1(5)
13. Edelman, B.J., Meng, J., Suma, D., Zurn, C., Nagarajan, E., Baxter, B.S., Cline, C.C., He, B. (2019). Noninvasive neuroimaging enhances continuous neural tracking for robotic device control. *Science Robotics*. 4 (31).
14. Iturrate, I., Antelis, J., Kubler, A., & Minguez, J. (2009). Non-Invasive Brain Actuated Wheelchair Based on a P300 Neurophysiological protocol and Automated Navigation. *IEEE*.
15. Wessels, M., Lucas, C., Eriks, I., and de Groot, S. (2010). Body Weight-Supported Gait Training for Restoration of Walking in People with an Incomplete Spinal Cord Injury: A Systematic Review. *Journal Compilation*. 42. 513-519.
16. Bayram, K.S., Kizrak, M.A., & Bolat, B. (2013). *Classification of EEG Signals by using Support Vector Machines*. IEEE.
17. Huang, X., Altahat, S., Tran, D., & Sharma, D. (2012). "Human identification with electroencephalogram (EEG) signal processing," 2012 *International Symposium on Communications and Information Technologies (ISCIT)*, Gold Coast, QLD, 1021-1026.

18. Exploring the Meaning of Math. (2018, May 7). MATLAB SVM *tutorial (fitcsvm)*. Retrieved from https://www.youtube.com/watch?v=q778MSq21vU.