

©Copyright 2024

Arsheya Raj

Innovative Rehabilitation Approach for Upper Limb Neurologic Conditions Using Mixed-Reality Simulation and EEG/EMG Biofeedback

Arsheya Raj

A thesis
submitted in partial fulfillment of the
requirements for the degree of

Master of Science

University of Washington

2024

Committee:

Erika Parsons

Kelvin Sung

Yusuf Pisan

Pierre Mourad

Program Authorized to Offer Degree:
Computer Science and Software Engineering

University of Washington

Abstract

Innovative Rehabilitation Approach for Upper Limb Neurologic Conditions Using
Mixed-Reality Simulation and EEG/EMG Biofeedback

Arsheya Raj

Chair of the Supervisory Committee:

Erika Parsons

School of Science, Technology, Engineering & Mathematics

Recent advances in Augmented Reality (AR), Mixed Reality (MR), Electroencephalogram (EEG), and Electromyogram (EMG) offer significant opportunities in medicine and neuroscience. This research aims to use these technologies to aid stroke patients with upper limb extremity weakness. This work extends the Edge Computing Ecosystem for Neuroscience Patients' Rehabilitation, part of the 'Stroke Rehabilitation Project' by University of Washington Bothell Engineering, University of Washington Seattle Neuroscience, and Rehabilitation Medicine at Harborview Medical Center (UWHM). Recently, UW Bothell's CSSE has also contributed, focusing on solutions for stroke patient rehabilitation. Traditional motor rehabilitation is costly, resource intensive, and often monotonous, reducing patient engagement. Using augmented and mixed reality technologies, interactive environments can be created on mobile devices, providing engaging and motivating experiences for patients. AR simulates real-world scenarios, offering a safe and fun way to practice tasks and aid in rehabilitation.

We used EEG and EMG sensors to conduct experiments and to collect data in a controlled environment targeting a reduced set of representative relevant motor tasks. The data were processed using various signal processing and statistical techniques, which in combination with the MR / AR simulation can be used to build a novel feedback and guidance system.

This system is a building block of our “NeuroRehab” ecosystem, which will use various ML models and algorithms for sequential prediction, with the aim of guiding patients through an optimal rehabilitation path within the 3D simulation environment.

Results showed that the combination of Frequency Filtering, ICA, and ERP with CNN model yield, so far, the best accuracy for classifying the motor tasks in EEG and EMG data. These findings contribute to the field of stroke rehabilitation of upper limb extremity weakness. It also contributes to a larger project that aims for better understanding and rehabilitation of other neurological ailments by offering insights to different hand gestures using EMG, and EEG data, and creating a framework for data processing, and feedback systems.

TABLE OF CONTENTS

	Page
List of Figures	iv
List of Tables	viii
Chapter 1: Introduction	1
1.1 Background	1
1.2 Goals and Motivation	4
1.3 Research Objectives	5
1.4 Scope and Limitations	8
Chapter 2: Related Work	9
2.1 Literature Review	9
2.2 Related Work	11
Chapter 3: Technical Background	16
3.1 Electroencephalography (EEG) Signals	16
3.2 Electromyography (EMG) Signals	21
3.3 Augmented/Mixed Reality	25
3.4 Upper Limb Extremity Weakness Background	26
3.5 Statistical and Machine Learning Background	28
Chapter 4: Methods	32
4.1 Experiment Design	32
4.1.1 Game Flow Sequences	33
4.2 Data Collection	37
4.3 Preprocessing Techniques	41
4.3.1 EEG Preprocessing	41

4.3.2	EMG Preprocessing	43
4.4	3D Simulation Design	45
4.5	Classification Models and Statistical Analysis	47
4.5.1	Statistical Analysis	50
4.6	Evaluation Metrics	51
4.7	Software and Tools	52
Chapter 5:	Results	54
5.1	Data Collection Results	54
5.2	Raw EEG and EMG Plot Analysis	54
5.3	Preprocessing Results	57
5.3.1	EEG Preprocessing Results	58
5.3.2	EMG Preprocessing Results	68
5.3.3	Anomalous Artifact Findings	70
5.4	Feature Engineering	70
5.5	Classification Results	71
5.5.1	Binary Classification	73
5.5.2	Multi-Class Classification	75
5.5.3	Statistical Analysis	81
5.6	Summary	84
Chapter 6:	Conclusion	86
6.1	Conclusion	86
6.2	Future Work	88
Appendix A:	Preprocessing Technique Configuration and PlugIns - EEGLAB	97
Appendix B:	Binary Classification Results	99
Appendix C:	3D Simulation Results	105
Appendix D:	Myo Connect - Myo Armband Manger	110
Appendix E:	DSI Streamer	114
E.1	Panels	114

E.2	Known Bugs	116
E.2.1	TCP/IP Program Slowdown/Speedup	116
E.2.2	Port Connection for the DSI Streamer	117
E.2.3	DSI Streamer Version	117

LIST OF FIGURES

Figure Number	Page
1.1 Edge Computing Ecosystem High-Level Architecture	2
1.2 Lateral hand-wave, formally known as Hand Flexion and Extension [39] . .	7
2.1 Schematic of experimental protocol based on two of six wrist and hand move- ments [48]	10
2.2 Architecture of our model showing training model for motion classification based on a CNN algorithm for each single datum [48]	11
2.3 MARS Invaders running on the iPad [26]	12
2.4 Topoplots of pre and post exercise for each sub-band [43]	13
2.5 A schematic representation of the three EMG experiments [34]	14
3.1 Examples of common brain waves in EEG data [53]	16
3.2 Distribution of EEG Electrodes [57]	17
3.3 EEG Signals using DSI-7 Streamer	18
3.4 Event-related potentials [30]	19
3.5 Cortical Homunculus Map [35]	20
3.6 Muscles of the forearm [9]	23
3.7 Diagram of MyoBand and its components [14]	23
3.8 Anterior and Posterior view of Upper Limb forearm [38]	24
3.9 Carpal Tunnel Syndrome physical exercise procedure flow chart [40]	27
3.10 Violin plot Distribution Explanation [22]	30
4.1 Hand Postures for experiments considered for this research. Lateral Hand- wave extension will informally be known as “Wave Out” and flexion as “Wave In”. Two-finger tap is also known as “Double Finger Tap” since it involves tapping twice with two fingers on the thumb.	34
4.2 Game Flow 1 ($\mathbb{G}_1$) [1]	36
4.3 Game Flow 2($\mathbb{G}_2$) [1]	36

4.4	DSI-7 Electrode Layout. LE is a reference node that acts as a conglomerate of the other electrodes. F3 and F4 would correspond to the brain's Frontal Lobe, while the rest of the electrodes are on Parietal Lobe.	38
4.5	Data Collection Diagram [13][56]. Data is wirelessly transmitted from the EEG headset to DSI Streamer, which runs on a PC, where it is stored in the hard drive.	39
4.6	Sample EEG Data in CSV format	40
4.7	Sample EMG and IMU Data in CSV format	41
4.8	EEG Data Preprocessing Pipeline	42
4.9	EMG Data Preprocessing Pipeline	44
4.10	Game Architecture	46
4.11	EEG data Classification and Statistical Analysis Layout	48
4.12	EMG data Classification and Statistical Analysis Layout	49
4.13	Confusion Matrix showing relationship between actual and predicted values in the form of true or false positive and negatives [21].	52
5.1	Plots of each Channel of Raw EEG Data collected for Game Flow 1 during one session using Matlab.	55
5.2	Each trend represents a each Channel of Raw EMG Data for one session of $\mathcal{G}_1$. Note how the signal shows disruptions compared to the baseline ($\mathcal{R}$) wave that starts at time 0.	56
5.3	Each axis of Raw IMU Accelerometer Data corresponds one session of $\mathcal{G}_1$ and time is captured in a heat-map fashion using the bar on the right.	56
5.4	4D Plots of each axis of Raw IMU Gyroscopic Data for one session of Game Flow 1.	57
5.5	Channel Locations in 2D and 3D	58
5.6	Plot of phase and magnitude vs. frequency after frequency filtering using a band pass filter of 8 – 13 Hz for a session in $\mathcal{G}_1$	59
5.7	ICA IC Label components for a session of $\mathbb{G}_1$	60
5.8	Selecting IC3 for a session of $\mathbb{G}_1$	61
5.9	Detail of IC3 for a session of $\mathbb{G}_1$. Top Left: Variability summary for the component IC3. Top Right: ICA Activity timeline. Bottom Left: Data Heatmap. Bottom Right: Activity Power Spectrum.	62
5.10	Detail of IC5 for a session of $\mathbb{G}_1$. Top Left: Variability summary for the component IC5. Top Right: ICA Activity timeline. Bottom Left: Data Heatmap. Bottom Right: Activity Power Spectrum.	63

5.11 ICA Component Maps 3D for one session in Game Flow 1	64
5.12 ICA - IC3 Component Maps in 3D for one session in Game Flow 1	64
5.13 Artifact Detection in Bin Based Epoched Data for one subject's session for Game Flow 1	65
5.14 ERPSet for one subject's session for Game Flow 1	66
5.15 ERP for channel F4 for one subject's session for Game Flow 1	66
5.16 ERP Scalp Maps for one subject's session for $\mathbb{G}_1$	67
5.17 Component ERPs with component map for one subject's session for $\mathbb{G}_1$. . .	68
5.18 3D Plots of each Channel of Filtered EMG Data for sample session for one volunteer in Game Flow 1. It is Frequency Filtered using a band pass filter of 20 – 450 Hz	69
5.19 EEG-ERP Features - Part 1	72
5.20 EEG-ERP Features - Part 2	72
5.21 EEG-ERP Features with Data Label	72
5.22 Overall EEG Features	72
5.23 EMG Features - Part 1	73
5.24 EMG Features - Part 2	73
5.25 EMG Features - Part 3	73
5.26 EMG Features - Part 4	73
5.27 IMU Features of Mean and Standard Deviation with Label	73
5.28 Overall EMG Features	73
5.29 Game Flow 1 ($\mathbb{G}_1$) ML Model Performance Comparison Violin Plot using EEG features only	76
5.30 Game Flow 2 ($\mathbb{G}_2$) ML Model Performance Comparison Violin Plot using EEG features only	76
5.31 Game Flow 1 ($\mathbb{G}_1$) ML Model Performance Comparison Violin Plot using EMG features only	78
5.32 Game Flow 2 ($\mathbb{G}_2$) ML Model Performance Comparison Violin Plot using EMG features only	78
5.33 Game Flow 1 ($\mathbb{G}_1$) ML Model Performance Comparison Violin Plot	80
5.34 Game Flow 2 ($\mathbb{G}_2$) ML Model Performance Comparison Violin Plot	81
5.35 Correlation Matrix Heat Map	82
5.36 Anova Analysis per feature group	83
A.1 EEGLAB opening window	97

A.2	EEGLAB Plugins Manager	98
C.1	Game Flow 1 scene in Unity	106
C.2	Game Flow 2 scene in Unity	106
C.3	Free Game Flow scene in Unity	107
C.4	Unity Profiler: CPU and GPU usage	108
D.1	Myo Connect-Armband Manager	110
D.2	Myo Connect-Custom Profile Calibration	110
D.3	Myo Connect-Make a fist	111
D.4	Myo Connect-Spread your fingers	111
D.5	Myo Connect-Wave In	112
D.6	Myo Connect-Wave Out	112
D.7	Myo Connect-Relax your arm	113
D.8	Myo Connect-Profile Calibration Complete.png	113
E.1	Data Source Panel in Streamer. Used to connect to DSI7 EEG Headset. . . .	114
E.2	Montage Panel in DSI Streamer. Used to calibrate each electrode on the DSI7 EEG Headset.	115
E.3	TCP/IP Panel in DSI Streamer.	115
E.4	How to avoid the TCP/IP Slowdown Bug. Do not disable TCP/IP during Data collection either.	116

LIST OF TABLES

Table Number	Page
4.1 Tasks considered for our experiments and their corresponding duration . . .	35
5.1 ML Models Metrics for Binary Classification	74
5.2 ML Models Metrics for $\mathbb{G}_1$ and $\mathbb{G}_2$ using EEG features only	76
5.3 ML Models Metrics for $\mathbb{G}_1$ and $\mathbb{G}_2$ using EMG features only	77
5.4 ML Models Metrics for $\mathbb{G}_1$ and $\mathbb{G}_2$ with combined features from EEG and EMG	79
B.1 ML Models Metrics for Rest vs Fist	99
B.2 ML Models Metrics for Rest vs Finger Spread	99
B.3 ML Models Metrics for Rest vs Hand Wave In	100
B.4 ML Models Metrics for Rest vs Hand Wave Out	100
B.5 ML Models Metrics for Rest vs Double Finger Tap	100
B.6 ML Models Metrics for Fist vs Finger Spread	101
B.7 ML Models Metrics for Fist vs Hand Wave In	101
B.8 ML Models Metrics for Fist vs Hand Wave Out	101
B.9 ML Models Metrics for Fist vs Double Finger Tap	102
B.10 ML Models Metrics for Finger Spread vs Hand Wave In	102
B.11 ML Models Metrics for Finger Spread vs Hand Wave Out	102
B.12 ML Models Metrics for Finger Spread vs Double Finger Tap	103
B.13 ML Models Metrics for Hand Wave In vs Hand Wave Out	103
B.14 ML Models Metrics for Hand Wave In vs Double Finger Tap	103
B.15 ML Models Metrics for Hand Wave Out vs Double Finger Tap	104

GLOSSARY

EDGE COMPUTING (EC) ECOSYSTEM: It refers to a distributed computing framework where data processing and analysis occur close to the data source, reducing latency and bandwidth use while enhancing real-time capabilities and efficiency.

AUGMENTED REALITY (AR): Augmented Reality (AR) overlays digital information onto the real world, enhancing the user's perception of their environment.

MIXED REALITY (MR): Mixed Reality (MR) blends real and virtual worlds, allowing digital and physical objects to interact in real-time.

AMYOTROPHIC LATERAL SCLEROSIS (ALS): It is a progressive neurodegenerative disease that attacks nerve cells in the brain and spinal cord, leading to muscle weakness.

PHYSICAL THERAPY (PT): PT is a treatment approach that uses exercises, manual techniques, and other interventions to improve movement, reduce pain, and enhance physical function.

SIGNAL-TO-NOISE RATIO (SNR): It measures the clarity of brain signals relative to background noise.

BRAIN COMPUTER INTERFACING (BCI): BCI enables direct communication between the brain and external devices.

MACHINE LEARNING (ML): It involves the development of algorithms that allow computers to learn from and make predictions based on data.

ELECTROENCEPHALOGRAPHY (EEG): A technique that records electrical activity of the brain via electrodes placed on the scalp.

ELECTROMYOGRAPHY (EMG): A technique which measures electrical activity in muscles to assess their function and health.

INERTIAL MEASURING UNIT (IMU): An IMU tracks motion and orientation using accelerometers and gyroscopes.

EVENT RELATED POTENTIAL (ERP): It measures brain responses triggered by specific events or stimuli.

INDEPENDENT COMPONENT ANALYSIS (ICA): It isolates distinct brain signal sources from mixed electrical activity.

FREQUENCY FILTERING (FF): A technique which isolates specific frequency ranges from a signal to remove unwanted noise or highlight relevant components.

FAST FOURIER TRANSFORM (FFT): Converts a signal from time domain to frequency domain for analysis.

ROOT MEAN SQUARE (RMS): It measures the average magnitude of a signal's variations.

MEAN ABSOLUTE VALUE (MAV): Calculates the average of the absolute values of a signal.

SUPPORT VECTOR MACHINE (SVM): It is a classification algorithm that finds the optimal boundary between classes by maximizing margin.

CONVOLUTIONAL NEURAL NETWORK (CNN): A type of deep learning model that excels in image and pattern recognition by using convolutional layers.

FEEDFORWARD NEURAL NETWORK (FNN): It is a type of neural network where information flows in one direction, from input to output, without loops.

RECURRENT NEURAL NETWORK (RNN): It is a neural network designed for sequential data, where connections form directed cycles to capture temporal dependencies.

LONG SHORT-TERM MEMORY (LSTM): A type of RNN that manages long-term dependencies by using gating mechanisms to control the flow of information.

ANALYSIS OF VARIANCE (ANOVA): It assesses differences between group means to determine if they are statistically significant.

KERNEL DENSITY ESTIMATE (KDE): In a violin plots, KDE shows the distribution of data by smoothing out its probability density.

INTERQUARTILE RANGE (IQR): In violin plots, the Interquartile Range (IQR) represents the middle 50% of the data, showing the range between the first and third quartiles.

EUROPEAN DATA FORMAT (EDF): A standard file format for storing EEG data.

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my thesis advisor, Prof. Erika Parsons, for her unwavering support, guidance, and invaluable insights throughout the entire process of researching and writing this thesis. Professor Parsons provided not only her academic expertise but also encouragement and motivation that were instrumental in the successful completion of this work. I would like to express sincere appreciation to Prof. Pierre Mourad, Prof. Kelvin Sung, and Prof. Yusuf Pisan for accepting my request to be on the committee for this thesis and for their valuable feedback. I am deeply grateful to my family, friends, and peers who supported me with their friendship, discussions, and encouragement throughout this journey. Their companionship made my academic experience enjoyable and meaningful.

Chapter 1

INTRODUCTION

1.1 *Background*

Recent advancements in Augmented Reality (AR), Mixed Reality (MR), Machine Learning (ML), Electroencephalogram (EEG), and Electromyogram (EMG) have allowed these technologies to be widely accessible, presenting numerous opportunities for applications, especially in medical fields such as neuroscience.

Electroencephalography (EEG) is a non-invasive technique used to record the electrical activity of the brain. EEG data is used in a wide range of applications, including clinical diagnosis, cognitive neuroscience, and brain-computer interfaces. Electromyography (EMG) is a technique used to measure and analyze electrical activity in muscles during contraction and at rest. It helps assess muscle function and diagnose neuromuscular disorders by detecting and recording electrical signals from muscles. Augmented Reality (AR) overlays digital information onto the real world to enhance the user's environment, while Mixed Reality (MR) blends real and virtual worlds, enabling interactive and immersive experiences, such as games.

In this research, we propose the combined use of these technologies to create and study new ways to help with the rehabilitation of patients recovering from neurological conditions such as neurological disorders and/or nerve damage in upper limb. The proposed work is an extension of the existing Edge Computing (EC) Ecosystem for Rehabilitation of Neuroscience Patients [46] [13], which was initially focused on developing solutions to help with the rehabilitation of stroke patients experiencing upper extremity weakness [5]. Figure 1.1 shows a high-level schematic of this ecosystem, focusing only on the EEG and EMG and AR/VR aspects.

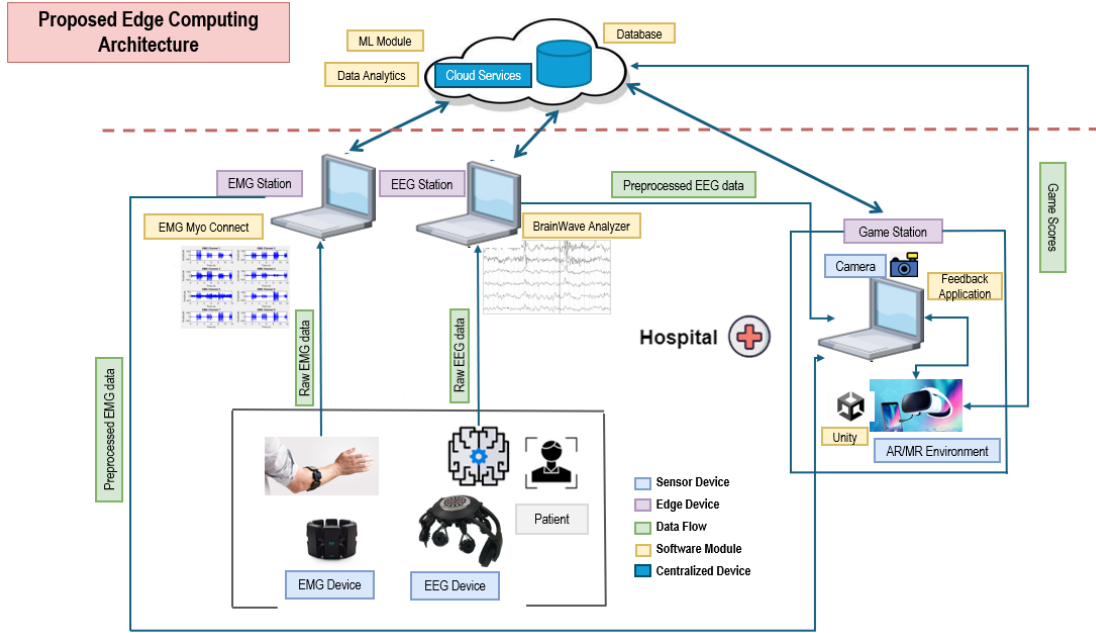


Figure 1.1: Edge Computing Ecosystem High-Level Architecture

Upper extremity weakness is a weakness in muscle strength in the arms, hands, and fingers. It can be unilateral (affecting one side of the body) or bilateral (affecting both sides of the body) [50]. It can also be focal (affecting specific muscles or muscle groups) or diffuse (affecting the entire upper extremity). Upper extremity weakness can be caused by a variety of factors, including:

- Neurological disorders, such as stroke, amyotrophic lateral sclerosis (ALS), multiple sclerosis, and spinal cord injury
- Nerve compression syndromes, such as carpal tunnel syndrome, and more
- Metabolic disorders, such as hypothyroidism and hyperparathyroidism

In particular, the scope of the proposed research will be the rehabilitation of upper extremity weakness caused by nerve injuries and spine compression. The goal of rehabilitation is to help people recover or adapt to the physical, cognitive, and emotional changes that

often result from neurological conditions [3].

Traditional motor rehabilitation, often therapist-driven and reliant on training equipment, typically occurs in rehabilitation centers under medical staff supervision. However, this approach incurs substantial costs in terms of manpower and resources. In addition, the repetitive nature of the training, which lacks effective patient-system interaction, often leads to monotony, diminishing the overall effectiveness of rehabilitation efforts. Many patients struggle to adhere to traditional Physical Therapy (PT) exercises [16] [47] [24], finding them tedious, challenging and / or boring, or may have a hard time commuting to hospitals or other locations where PT is performed.

To address these challenges, Mixed Reality (MR) and Augmented Reality (AR) technologies offer a promising solution. AR and MR technologies enables the creation of interactive and immersive environments on personal mobile devices and laptops. This can enable the creation of rehabilitation-related games that can help improve patient engagement and motivation during therapy exercises. In addition, the use of these technologies can allow feedback and guidance that can support patients in improving their motor skills and cognitive abilities. AR can simulate real-world scenarios and games on portable devices, allowing patients to practice functional tasks and regain independence in a more comfortable, accessible, yet controlled environment [23].

In this study, Thalmic Labs' MyoBands were used to capture EMG data and hand gestures[15] . The Myo armband, also known as the "MyoBand," is a wearable technology designed to capture Electromyogram (EMG) signals and Inertial Measuring Unit (IMU) Motion data emanating from the forearm muscles. EEG can be used to capture information about brain pathology through the analysis of neural activity patterns. EEG can be used to measure event-related potentials (ERP), which are changes in electrical activity in the brain in response to specific events or stimuli. Classification of motor tasks from EEG data has been the subject of extensive research in recent years. Various preprocessing techniques, such as frequency filtering (FF), and independent component analysis (ICA) have been employed to increase the signal-to-noise ratio (SNR) of EEG data and improve classification accuracy.

EEG data contains a significant amount of noise, including EMG contamination caused by the propagation of muscle signals into EEG data. Reducing this noise is of significant interest to brain computer interfacing (BCI).

In addition, different classification models, including support vector machines (SVM), feedforward neural networks (FNN), convolutional neural networks (CNN), recurrent neural networks (RNN), long-short-term memory (LSTM), were used to classify motor tasks from EEG and EMG data. Evaluation metrics such as gesture recognition accuracy, F1 score, precision and recall have been used to assess the performance of these classification models.

1.2 Goals and Motivation

The motivation of this research is to explore the gamification of neurological conditions rehabilitation to create a more engaging and enjoyable experience for patients. In particular, our work explores the design of AR/MR simulations with the aim of improving upper limb extremity stroke rehabilitation. This study will examine the AR/MR simulation’s design platform, content, and mobile application design, which in combination with data analysis, ML, and the incorporation of related medical advice, will provide a foundation for the development of similar tools for other neurological conditions. Furthermore, this work will be an initial component of a larger Edge Computing (EC) ecosystem, which envisions the use of similar approaches and technologies to enhance recovering neurology patients. This system is dubbed the “Smart NeuroRehab Ecosystem”.

Our motivation, from a community-oriented perspective, is to create a more engaging, accessible, modernized, and technologically-driven tool for specialized rehabilitation therapies that supports both, patients and medical professionals:

- From a patient’s perspective, the need for a more engaging and interactive way to carry out rehabilitation therapy exercises even outside a hospital setting.
- To provide a tool for rehabilitation and medical professionals that allows them to track their patients progress and eventually help create “smart” exercise routines for patients.

From a technical perspective, our motivation is to study the combined use of emerging technologies (both hardware and software) in a medical application setting. Particularly, to bring in statistical and ML techniques to analyse EEG and EMG data to study emerging patterns that can then be used to create adaptive gaming routines. Routines that with the use of AR/MR, could be used to transform dull tasks into entertaining tasks with flexible challenge levels. In Section 1.3 below, we will provide specifics of our research objectives.

The concept of the “Smart NeuroRehab Ecosystem” was envisioned by Dr. Parsons, motivated by the existing “Stroke Rehabilitation Project”, started by the Engineering Department at the University of Washington Bothell (UWB) in conjunction with UW Seattle Neuroscience and Rehabilitation Medicine at Harborview Medical Center in Seattle (UWHM) and more recently CSSE (Computer Science and Software Engineering) (UW Bothell).

The research presented in this thesis encompasses novel groundwork for the NeuroRehab Ecosystem, and it is influenced by and extends from the aforementioned Stroke Rehabilitation Project.

1.3 Research Objectives

The broader objective of this work is to create a proof-of-concept AR/MR game/simulation capable of recognizing hand gestures that can assist therapists in the task of rehabilitating patients with upper limb neurological issues. In our work, we will refer to this novel type of adaptive software application as “NeuroGame”¹. Its development also involves the integration of MyoBands to capture muscle signals, which introduce a new dimension of interactivity and functionality to improve the AR/MR experience. This is a first of more potential NeuroGames that involve the use of biofeedback devices, all part of the NeuroRehab Ecosystem mentioned in 1.2. Patients’ ethical and privacy concerns have been considered for this system but are out of the scope of this thesis work [46].

Since this research involves the combination of various technologies, we can group the

¹The term *neurogames* has been loosely used to refer to a childrens’ educational platform as described in Wikipedia [59], however, this is unrelated to this thesis work.

objectives of this research into the following categories:

- Game/Simulation Development: involving the use of Augmented and Mixed Reality
- ML and HPC: also encompassing Data Analysis to study EEG and EMG data and model creation and training
- Medical Application Domain: the use of Neuroscience concepts, devices, and rehabilitation therapy

It is worth mentioning that the categories of game development or 3D simulation and medical application are heavily intertwined since the initial motivation of this work was *gamification* of stroke rehabilitation therapy.

Next, we detail specific objectives based on the intersection of these categories:

Game Development and Rehabilitation

- Explore the integration of an AR/MR application for *gamifying* rehabilitation sessions. Collaboration with neuroscience experts along the way will ensure the value and effectiveness of the game.
- Provide an immersive experience for patients, while simultaneously simplifying data collection and monitoring for medical professionals. The scoring system of the game will reflect individual progress and goals within personalized profiles, with the purpose of improving patient engagement and motivation. Feedback and guidance further contribute to motor-skill relearning and goal attainment.
- Create a basic flexible gaming platform for personal devices. Although originally intended for hospital deployment, through the use of a smartphone, PC or tablet, patients could continue rehabilitation at home, similar to experiences like Pokémon Go [29].

Machine Learning and HPC

- Investigate the effectiveness and performance of various data preprocessing techniques for the classification of motor tasks in EEG and EMG data, e.g., FFT, ICA, ERPs. We

focus on a set of motor tasks: making a fist, finger spread, hand flexion and extension and double finger tap (see Figure 1.2).

- Develop an experimental design for data collection: as part of the EC environment, we incorporate the use of various types of sensors, e.g., EEG and EMG devices, and cameras for face and hand tracking (used in different parts of the larger NeuroRehab project). This is a crucial aspect of this research as EEG data is complex and much is still not well understood, so we need an environment where we can control as many variables as possible and aim for result repeatability.
- Develop our application as part of an Edge Computing (EC) environment (the NeuroRehab Ecosystem depicted in Figure 1.1). AR/MR and ML can be computationally expensive, and EC environments are subject to resource constraints – diverse types of computational devices are being used, from microprocessors to large servers working together, which in addition, can make deployment challenging.

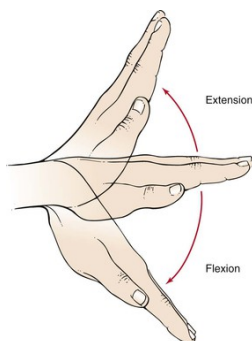


Figure 1.2: Lateral hand-wave, formally known as Hand Flexion and Extension [39]

Medical Application Domain

- **Neuroscience Concepts:** Apply motor learning and neuroplasticity principles to create effective rehabilitation tools. In rehabilitation, neuroplasticity refers to the

brain's capacity to reorganize and form new neural pathways to recover lost functions and adapt to new conditions following injury or stroke.

- **Rehabilitation Devices:** Utilize MyoBands and EEG headsets for muscle and brain activity monitoring to enhance therapy.
- **Therapeutic Goals:** Design the AR/MR simulation to support personalized therapy for conditions like stroke and carpal tunnel syndrome, and collaborate with therapists to validate and refine features.

1.4 Scope and Limitations

While conducting this study, several limitations surfaced, and I would like to transparently acknowledge them :

- EEG signals contain a lot of noise so it will create problems when we train the ML models.
- It is worth mentioning that Thalmic Labs' Myo Software is no longer supported by the company. This means acquiring the software may be difficult, and if it isn't pre-installed on your computer, you might need to look for alternative solutions, such as finding a public repository.
- The EEG headset was not in optimal condition during the experiments.
- Data for the experiments is collected from University of Washington Bothell student volunteers, this potentially influences generalizability to other groups. The experiments do not include hyperparameter tuning for preprocessing or machine learning models, which affects the generalizability of the findings.

The next chapter explores related works in AR/MR games for stroke rehabilitation, incorporating Brain Signals, EMG Signals, which are similar and different from this study.

Chapter 2

RELATED WORK

In this chapter, the existing work related to Augmented Reality/Mixed Reality Game Designs, Electromyography (EMG) and Electroencephalography (EEG) signals that support our research and study is discussed.

2.1 Literature Review

Brain-Computer Interface (BCI) training complements traditional physical therapy and may improve upper extremity motor function and brain recovery in stroke patients [4]. The BCI-based device and its associated therapy align with the principles of classical neuromotor rehabilitation. It offers physical therapists a tool to monitor and evaluate the patient's involvement in cognitive exercises, while also supporting patients in practicing motor tasks [20].

An AR game can offer interactive immersive exercises tailored to upper limb weakness rehabilitation, with tasks that mimic real-world movements to encourage engaging and enjoyable activities and games [26]. EEG, which measures brain activity using a BCI, can be incorporated to enhance the adaptability of the AR game. By analyzing EEG signals, the game can dynamically adjust difficulty levels, pace, and task complexity based on the user's cognitive responses. This creates a personalized and responsive gaming experience [43]. EMG, which measures muscle activity, adds another layer of feedback. The game can use EMG data to assess muscle engagement during specific tasks. This information helps to provide feedback on the effectiveness of movements performed in the game, guiding users to optimal muscle activation for rehabilitation [34].

These technologies can be combined, as detailed in the research paper of “Enhanced

recognition of amputated wrist and hand movements using deep learning and multimodal fusion of EMG and EEG” [48]. Data were collected from five amputees and nine healthy controls according to the experiment protocol shown in Figure 2.1. This research paper talks

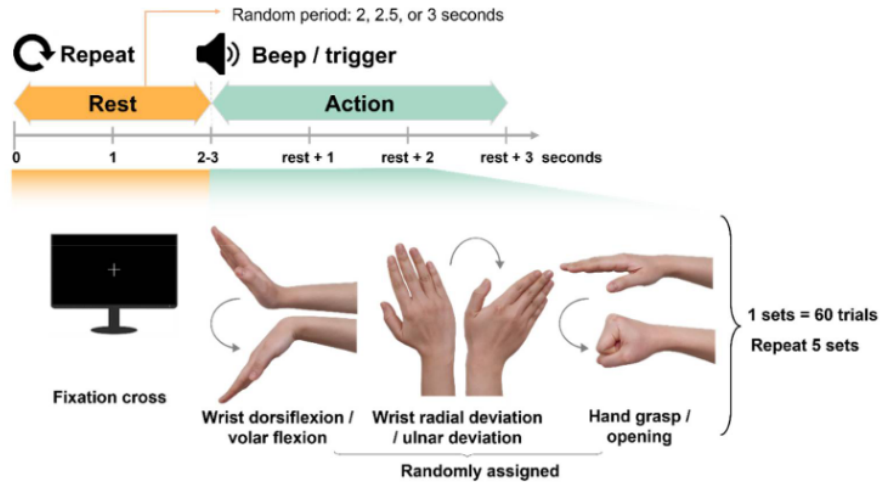


Figure 2.1: Schematic of experimental protocol based on two of six wrist and hand movements [48]

about motion classification for prosthetic arm control using EEG and EMG signals is limited by noise and resolution issues. This study proposes a system fusing these signals into 2D images for transfer learning, improving accuracy by 4.18-4.35 percent in healthy controls and 2.51-3.00 percent in transforearm amputees.

It explains how to combine and process EEG and EMG signals, helping us understand how to use these data together to create a better rehabilitation system and showing why it's important.

This research was done in collaboration with **Sunjl Gahatraj**, a fellow Master's student from the University of Washington Bothell, CSSE. Sunjl's project, titled "Enhancing Neurological Rehabilitation: Combining Gaming, Robotics, and Machine Learning in an Edge Computing Environment," is quite similar to mine. While Sunjl focused on using robotics

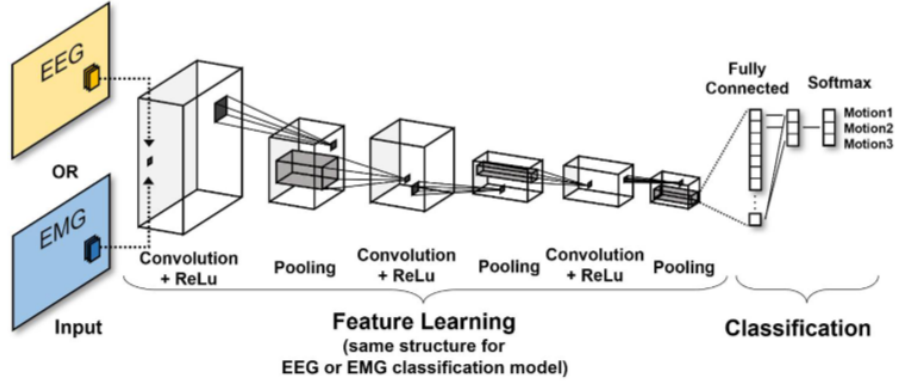


Figure 2.2: Architecture of our model showing training model for motion classification based on a CNN algorithm for each single datum [48]

and camera input to detect hand movements, I used AR/MR environments and EMG data for simulation. We worked together on developing the EEG data pre-processing pipeline and identifying tendon and nerve glide exercises. Sunjil also contributed to parts of the binary classification code.

Several recent studies have addressed the topic of Neurological Rehabilitation for Upper Limb extremity weakness using AR and EEG and/or EMG. Some of the recent related works are as follows:

2.2 Related Work

Augmented and Mixed Reality

One of the research papers titled as “The Mobile RehApp(TM): an AR-based mobile game for ankle sprain rehabilitation” by Garcia and Navarro discusses Mobile 3D AR games[26]. Recognizing the common issue of people losing interest in rehabilitation exercises due to their repetitive nature, there have been recent efforts to incorporate interactive video games into rehabilitation programs. These innovative approaches, exemplified by successful implementations such as Nintendo Wii and Microsoft Kinect games, have shown positive results in

engaging users in physical activities while making the rehabilitation process more enjoyable. Surprisingly, there is a noticeable gap in the market when it comes to portable rehabilitation tools in the form of interactive games.

This paper introduces the Mobile RehApp™, an augmented reality-based application tailored for mobile devices as shown in Figure 2.3 . This groundbreaking tool is designed to provide therapeutic support, specifically targeting ankle sprain rehabilitation. The aim is to



Figure 2.3: MARS Invaders running on the iPad [26]

not only enhance patient engagement but also offer a valuable resource for physiotherapists, revolutionizing the landscape of rehabilitation through the integration of interactive and portable technologies.

This research paper demonstrates the usefulness of an AR game in the mobile / ipad version and helps us understand how a portable game is useful for rehabilitation. It also discusses the limitations of these games, like these games require either an immovable object, rubber tubing, weights, or the clinician to provide resistance, or it requires a board in which the user stands and keeps from falling. Additionally, these games can be developed in mixed reality (MR) to provide a more immersive experience by combining features of both AR and VR.

Electroencephalography (EEG)

A different study which is titled as “Wavelet the EEG signals used for Stroke Rehabilitation” by Singh discusses EEG signals post-stroke[43]. Post-stroke monitoring is crucial for understanding patients’ progress during rehabilitation. The study focuses on evaluating the effectiveness of rehabilitation exercises by investigating various parameters that play a role in observing immediate changes. Over a 90-day period, different wavelet coefficients were used to extract parameters, with energy and waveform length showing significant variation before and after the exercise. The study also conducted region-specific analyzes to monitor changes in different parts of the brain, providing a comprehensive understanding of post-stroke recovery.

It demonstrates how EEG signals can help establish a recovery system after stroke. This research paper is interesting as it compares the data for Pre and Post the exercises as shown in Figure 2.4.

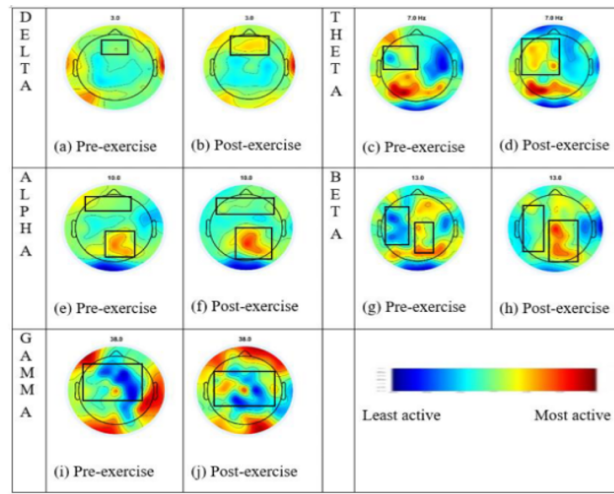


Figure 2.4: Topoplots of pre and post exercise for each sub-band [43]

Most research papers usually discuss post-stroke results only with 'rest' movement. It also offers a valuable perspective to improve task design in mobile games, particularly to

optimize stroke recovery.

Electromyography (EMG)

Another research article titled “A VR-Based Motor Imagery Training System With EMG-Based Real-Time Feedback for Post-Stroke Rehabilitation” discusses the role of combining VR and EMG in Stroke Rehabilitation by describing how motor tasks that use EMG are combined with a VR environment[34]. In the realm of post-stroke rehabilitation, a VR-based motor imagery system is making strides. Addressing the challenge of confirming patient engagement during motor imagery training, this system incorporates surface electromyographic (EMG) real-time feedback. In experiments involving post-stroke patients, the VR system immerses them in a basketball shooting scenario for bilateral upper limb training as shown in Figure 2.5. EMG data, collected during the evaluation, is used to personalize the training and quantitatively assess the degree of participation.

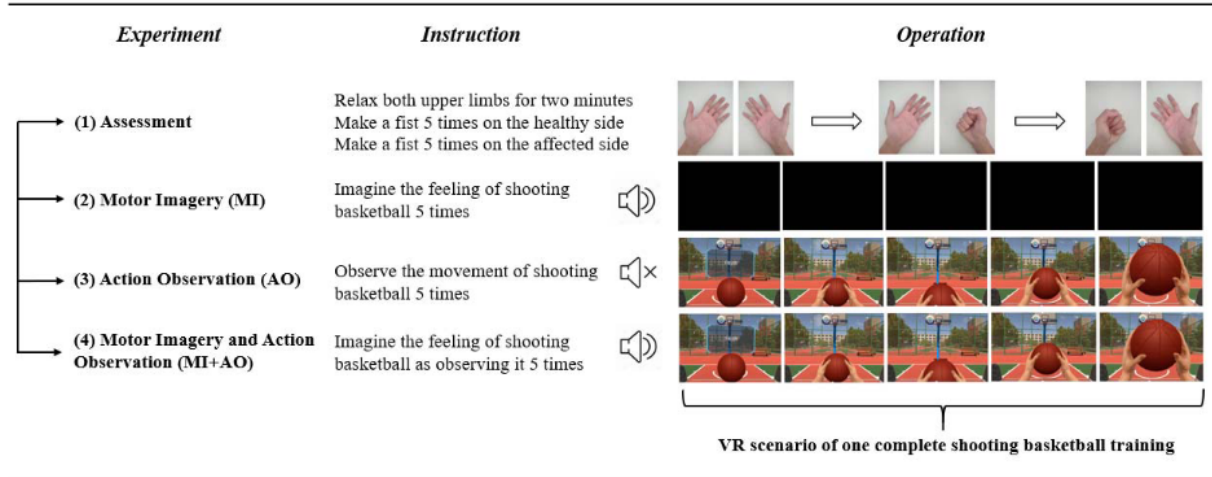


Figure 2.5: A schematic representation of the three EMG experiments [34]

Results indicate significant differences in muscle strength between different experiments, showcasing the effectiveness of EMG-based feedback. This VR-assisted motor imagery sys-

tem emerges as a promising avenue for personalized and engaging post-stroke rehabilitation.

The discussion of a basketball game in this paper is not only relevant to my upper limb extremity stroke research but also holds potential for incorporating hand gestures in the game. The setup of the experiment, which involves data from both adults and children, is particularly valuable for my EMG analysis. This adds a dynamic layer to the research, offering insights that bridge both the rehabilitation context and the interactive elements of a basketball game. These interconnected insights pave the way for a holistic and purposeful approach in my research on game design and rehabilitation.

Furthermore, a highly relevant research paper was titled “Hand Gesture Recognition Using EMG-IMU Signals and Deep Q-Networks” [27]. It explores hand gesture recognition systems using EMG and IMU (Inertial Measuring Unit) signals, often leveraging supervised machine learning. This work introduces a reinforcement learning (RL) approach with a deep Q-network (DQN) to classify and recognize eleven hand gestures, achieving up to 98.95% accuracy with the Myo armband sensor. It demonstrates how combining EMG data with IMU data can enhance the training of ML or Deep Learning (DL) models.

Chapter 3

TECHNICAL BACKGROUND

In this chapter we discuss the technical background for this work related to motor classification of EEG Signals and EMG Signals, Mixed Reality (MR)/Augmented Reality (AR), and statistical analysis/machine learning.

3.1 *Electroencephalography (EEG) Signals*

Electroencephalography (EEG) is a technique that records electrical activity in the brain through electrodes placed on the scalp. Brain-computer interfaces (BCI) use these EEG signals to interpret brain activity and facilitate direct communication between the brain and external devices, enabling the control of computers, prosthetics, or other assistive technologies [33]. Brain waves primarily consist of alpha, beta, theta, and delta waves, with less common waves being gamma and mu waves. Figure 3.1 below shows some examples of common brain waves. These waves are primarily measured by frequency and amplitude. Further, each wave is associated with a different mental and physical state, for example, alpha waves are associated with a restful state with little mental activity, while beta waves are associated with intense activity or concentration.

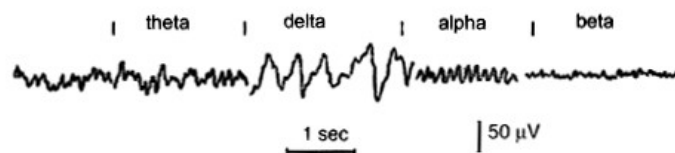


Figure 3.1: Examples of common brain waves in EEG data [53]

Currently, the most prevalent method for monitoring brain states is electroencephalography (EEG). An EEG device measures cortical brain activity by detecting electrical potentials generated by synapses that activate on the surface of the brain [32]. The data from an EEG are recorded as a waveform in microvolts and time in an electroencephalogram. Figure 3.2 shows the distribution of the EEG electrodes.



Figure 3.2: Distribution of EEG Electrodes [57]

The Wireless DSI-7 of Wearable Sensing is the leading dry electrode EEG system in terms of signal quality and comfort. We used the DSI Steamer for the data collection from the headset. Figure 3.3 displays the default screen of the DSI Steamer after connecting to the EEG headset [56]. The DSI Steamer transmits physiological data from a monitoring system, such as EEG signals, to a computer via a COM port.

Independent Component Analysis

Independent Component Analysis (ICA) is a powerful signal processing technique applied in EEG data analysis to decompose complex mixed signals into their underlying independent components. EEG signals are a blend of neural activity, physiological artifacts (such as eye

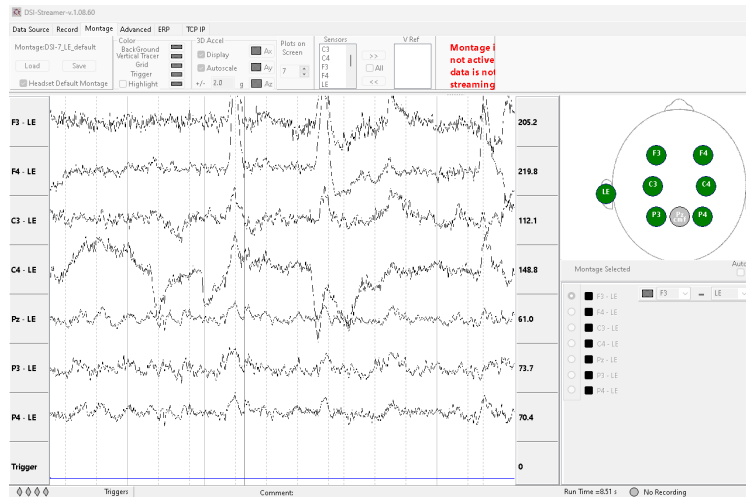


Figure 3.3: EEG Signals using DSI-7 Streamer

blinks and muscle movements), and environmental noise. ICA leverages the assumption that these components are statistically independent to effectively separate and isolate each source of signal.

This decomposition allows us to pinpoint and examine specific neural signals associated with cognitive processes or motor tasks, which might otherwise be hidden by noise or artifacts. By isolating and removing unwanted artifacts, such as those of eye movements or muscle contractions, ICA improves the clarity and interpretability of EEG data. This improvement facilitates a more accurate analysis of brain activity, contributing to better understanding of cognitive and neurological functions [10].

Frequency Filtering

Frequency filtering (FF) is a common technique in motor task preprocessing for both EEG and EMG signals, designed to isolate specific frequency ranges of interest. This method enhances the signal-to-noise ratio by amplifying certain frequency bands, thus improving the clarity of relevant neural activity. Various filters, such as high-pass, low-pass, band-pass or notch filters, can be used depending on the research goals and the characteristics

of the EEG and EMG data, with band-pass filtering being the most widely used during preprocessing [58]. This approach enables researchers to focus on specific frequency ranges linked to different brain wave activities, such as alpha, beta, theta, or gamma waves, allowing for a more focused analysis of brain dynamics and functional connectivity.

Event Related Potentials

Event-related potentials (ERPs) in EEG are brainwave patterns that occur in response to specific sensory, cognitive, or motor events. They are used to study how the brain processes stimuli and performs tasks. ERPs stem from sensory or motor-evoked potentials and can vary in intensity and timing. Examples include making a fist, pinching, touching the face, hand waving, and many more. Furthermore, brain waves differ with age, with certain types more common in children than in adults, and vice versa.

Figure 3.4 displays the ERP waveform and its features. The ERP waveform represents brain responses to specific stimuli, with features like N100, N200, P200, and P300 reflecting different stages of cognitive processing. N100 and N200 are negative deflections related to early sensory and cognitive processing, while P200 and P300 are positive deflections associated with higher cognitive functions such as hand movements and decision-making [41] [42].

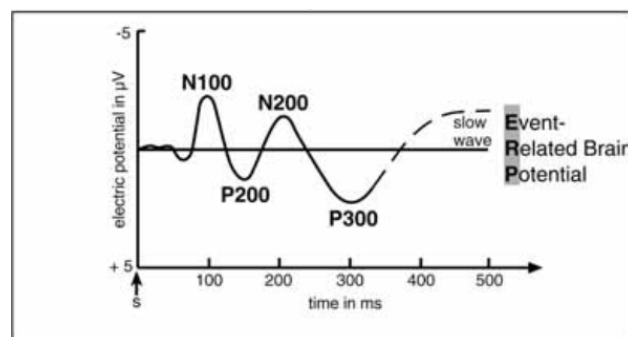


Figure 3.4: Event-related potentials [30]

Skewness in ERP Data measures how the distribution of ERP values deviates from symmetry. Positive skewness indicates a longer tail on the right, suggesting more high values, while negative skewness indicates a longer tail on the left, suggesting more low values. A skewness close to zero suggests a symmetrical distribution of ERP responses.

Kurtosis in ERP Data assesses the "tailedness" of the distribution. High kurtosis indicates heavy tails and a sharp peak, meaning frequent extreme values, while low kurtosis indicates lighter tails and a flatter peak, suggesting fewer extreme values. Kurtosis close to zero implies a standard distribution with typical levels of extreme responses.

Figure 3.5 below shows the mapping of senses and motor actions to the brain cortex. It is a cortical homunculi, also known as sensory or motor homunculi, is a visual map of the body on the brain's cortex, showing distorted proportions based on the body's representation in the somatosensory and motor areas.

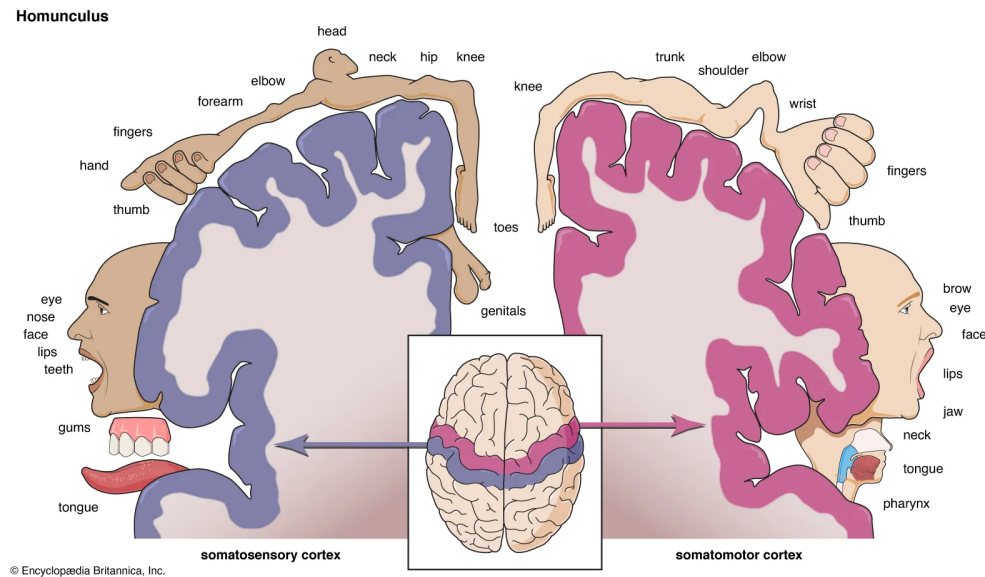


Figure 3.5: Cortical Homunculus Map [35]

EEGLAB is a widely-used open-source MATLAB toolbox designed for processing, analyzing, and visualizing electroencephalographic (EEG) data [7]. It offers a comprehensive

suite of tools for tasks such as data preprocessing, artifact rejection, independent component analysis (ICA), and event-related potential (ERP) analysis. EEGLAB is user-friendly and extensible, allowing researchers to implement custom scripts and plugins for specialized analyses.

Next, we formally define the concept of *artifact*, which is used throughout this thesis and in the context of EEG signals:

Definition 1. An **artifact** is an unwanted electrical activity or noise that originates from sources other than the brain. These can include muscle movements, blinks, heartbeats, and external electrical interference.

Artifacts can distort the EEG data, making it difficult to accurately interpret the underlying neural activity, so a means to remove them should be in place. An **Artifact Rejection** refers to the process of identifying and removing or correcting these unwanted signals from the EEG data to improve its quality and accuracy. Techniques for artifact rejection include manual inspection, automated algorithms, and methods like independent component analysis (ICA) to isolate and eliminate the non-neural components from the EEG recordings.

3.2 *Electromyography (EMG) Signals*

The Myo armband, often referred to as the “MyoBand,” is a wearable technology which is designed to capture electromyography (EMG) signals and motion data from the forearm muscles and interpret them into gestures and movements. The Myo armband allows users to control digital devices, interact with virtual and augmented reality environments, and perform actions through intuitive hand and arm movements. Figure 3.7 unveils the composition and components of the MyoBand.

In this study, Thalmic Labs’ Myo Armband was used to collect data using Myo Connect software. Each Myo Armband is equipped with eight sensors, each providing a channel for data collection. These channels help to capture detailed muscle activity, which can be useful for applications such as gesture recognition and rehabilitation exercises. The Myo Armband

is placed around a subject's forearm, however, the specific muscle groups each channel tracks can vary depending on the placement and orientation of the device. The following are some of the muscles that can be tracked with this hardware (see Figure 3.6 for visual reference of muscle location on the human arm):

- Channel 1: Extensor carpi radialis
- Channel 2: Extensor digitorum
- Channel 3: Extensor carpi ulnaris
- Channel 4: Flexor carpi ulnaris
- Channel 5: Flexor digitorum superficialis
- Channel 6: Flexor carpi radialis
- Channel 7: Pronator teres
- Channel 8: Brachioradialis

Figure 3.8 highlights the muscle groups tracked by the Thalmic Labs Myobands electrodes. It displays 4 muscle groups from the anterior view and 4 from the posterior view of the upper arm, all tracked during our experiments using the Myobands.

Zero Crossing Rate (ZCR): This metric counts how often the EMG signal crosses the zero line, offering insights into the frequency of muscle contractions and the overall complexity of the signal.

Waveform Length (WL): This feature measures the total length of the EMG signal's waveform, reflecting the complexity of muscle activity by capturing the extent of fluctuations over time.

Inertial Measuring Unit (IMU)

In EMG studies, an Inertial Measurement Unit (IMU) is used to track the movement and orientation of the body part being analyzed. This integration improves the interpretation of muscle activity by providing valuable context on the physical position and movement of the body [27].

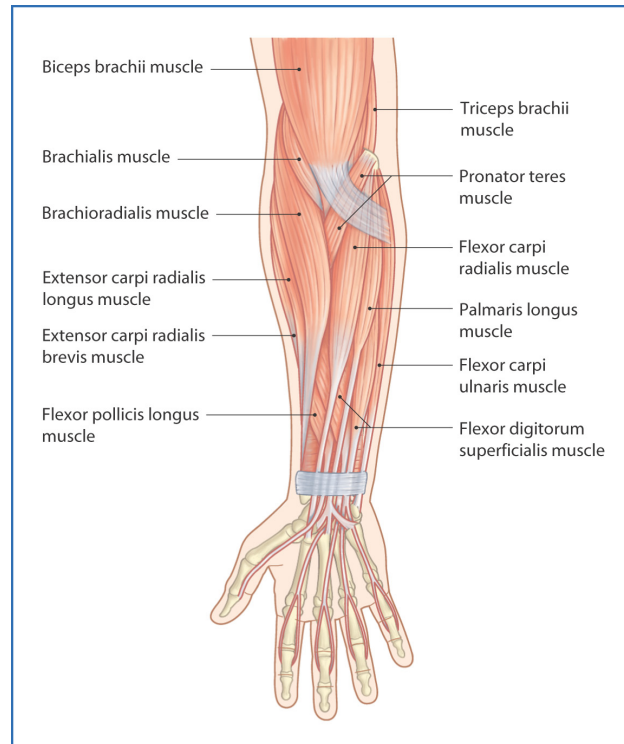


Figure 3.6: Muscles of the forearm [9]

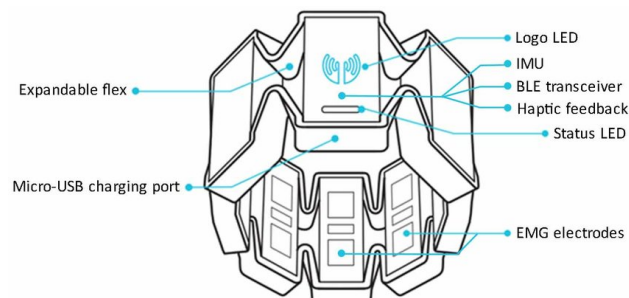


Figure 3.7: Diagram of MyoBand and its components [14]

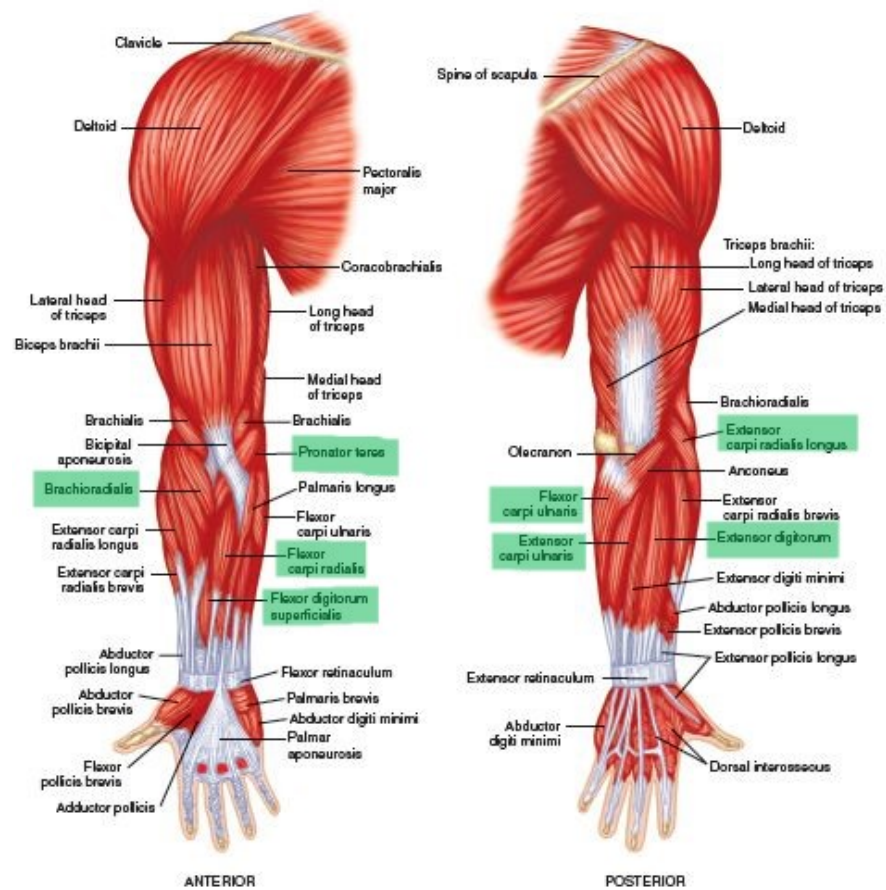


Figure 3.8: Anterior and Posterior view of Upper Limb forearm [38]

The IMU consists of two primary sensors:

- **Gyroscopic Data:** Measures rotational movement and changes in orientation by detecting angular velocity. This data helps to understand how the body part rotates or tilts during muscle contractions.
- **Accelerometer Data:** Measures linear acceleration and changes in velocity. It captures how quickly the body part speeds up or slows down, providing insights into translational movements.

By combining gyroscopic and accelerometer data, the IMU offers a comprehensive view of the movement dynamics of the body part. This detailed information improves the accuracy of muscle activity interpretation and provides additional features for training machine learning models, leading to enhanced performance and precision in motion analysis and rehabilitation applications.

3.3 Augmented/Mixed Reality

Augmented Reality (AR) offers a transformative impact on upper limb extremities neurological disorder rehabilitation [60]. By providing immersive visualizations of anatomical structures and interactive exercises, AR enhances patient engagement and understanding. Real-time feedback during exercises, personalized interventions, and remote support possibilities through AR empower both patients and healthcare professionals. We believe that the ability of technology to turn rehabilitation into interactive experiences not only motivates patients but also allows for tailored rehabilitation plans, fostering more effective and enjoyable recovery journeys. AR's potential in data collection and analysis further contributes to refining rehabilitation strategies, promising more personalized and impactful outcomes in stroke rehabilitation.

Unity 3D is a widely used game development engine that supports 2D, 3D, augmented reality (AR) and mixed reality applications. It offers cross-platform development, advanced

graphics, and an extensive asset store.

Mixed Reality (MR) combines virtual reality (VR) and augmented reality (AR) by integrating virtual objects into the real world, allowing users to interact with both digital and physical elements at the same time. In MR, virtual objects blend with real-world surroundings, letting users perceive and interact with them as if they were actually there. This technology ranges from fully immersive virtual experiences to subtle enhancements of the real world.

3.4 Upper Limb Extremity Weakness Background

Upper limb extremity weakness refers to a reduction in strength and function in the muscles of the arms, which can result from various conditions such as stroke, nerve injuries (neurological), or muscular disorders. This weakness can significantly impact daily activities, making tasks like lifting, gripping, and reaching challenging. Upper limb extremity weakness, from a neurological perspective, often stems from disruptions in the nervous system that affect muscle control and strength.

The following exercises are commonly part of rehabilitation therapy for upper limb weakness and can help improve strength, flexibility, and coordination in the hands and wrists, which is crucial for daily activities [44][36]:

1. Grip Strengthening:

- **Squeeze Ball:** Squeeze a soft ball for 5-10 seconds, then release. Repeat 10-15 times. It is similar to making a fist.

2. Finger Dexterity:

- **Finger Taps:** Tap each finger on the thumb, one at a time, as quickly as possible.

3. Wrist Flexibility:

- **Wrist Flexion and Extension:** Bend the wrist forward and backward, holding each position for a few seconds.
- **Wrist Rotations:** Rotate the wrist in circular motions, both clockwise and counterclockwise.

4. Range of motion:

- **Finger Stretches:** Spread fingers wide apart and then bring them back together.
- **Thumb Opposition:** Touch the thumb to each fingertip, one at a time.

Various PT exercises are designed to treat upper limb extremity weakness. A particularly effective method is tendon and nerve glide exercises. These exercises are based on the hand exercises described in Section 3.4. Figure 3.9 shows several exercises commonly used in PT regimens to treat carpal tunnel syndrome and general upper limb extremity weakness [40], these were used as a model for the set of exercises used in our experiments.

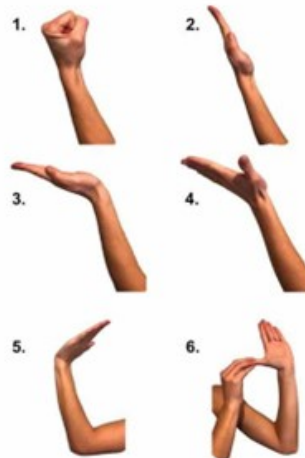


Figure 3.9: Carpal Tunnel Syndrome physical exercise procedure flow chart [40]

The exercises selected for our experiments vary from these, but are more easily tractable by our EMG device (the MyoBand).

3.5 Statistical and Machine Learning Background

Fundamental Statistical Measures

In statistics, several fundamental measures describe data groups, including the mean, variance, and standard deviation. While other basic measures exist, they are not covered in this work. These values help to describe the centrality and dispersion of the data groups and are defined below.

Mean: The average of a set of numbers, calculated by dividing the sum of the numbers by their count. Also known as the arithmetic mean, it is the most commonly used measure of central tendency.

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i = \frac{x_1 + x_2 + \dots + x_n}{n} [6] \quad (3.1)$$

Variance: A measure of dispersion that describes how spread out the values in a data set are around the mean. It is calculated as the average of the squared differences from the mean and is also known as the square of the standard deviation.

$$\text{Var}(X) = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2 \quad (3.2)$$

Standard Deviation: A measure of dispersion that indicates how much the values in a data set deviate from the mean. It is expressed in the same units as the original data, making it intuitive and useful for understanding variability. It is sensitive to outliers and aligns with other statistical measures, providing a clear sense of the data's spread.

$$\text{SD}(X) = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2} \quad (3.3)$$

Mean Absolute Value (MAV): A measure of the average magnitude of a signal by computing the mean of the absolute values of its samples, indicating the overall level of the

signal.

$$\text{MAV}(X) = \frac{1}{n} \sum_{i=1}^n |x_i| \quad (3.4)$$

Root Mean Square (RMS): A measure of the square root of the average of the squared values of a signal, reflecting its power and intensity, and providing insight into the signal's strength and variability.

$$\text{SD}(X) = \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \quad (3.5)$$

Analysis of Variance

To evaluate the level of significance and reject the null hypothesis where multiple independent groups exist would require a large number of t-tests, therefore this work utilizes a process of the analysis of variance (ANOVA) test. ANOVA seeks to evaluate statistical power on all groups simultaneously and thus describes how closely grouped all the groups are. It accomplishes this using sum of squares to calculate variances across groups. Breaking down the sum of squares components into totals, effects, and errors allows ANOVA to generate a F-statistic value and generate a p-value for all the groups based on that value. The equations for the sum of squares of error (also known as the residual sum of squares) and the F-statistic are below. In equation 3.6 k is the number of groups, n_i is the number of observations in the i - th group, Y_{ij} is the j - th observation in the i - th group, and $\bar{Y}_i$ is the mean of the i - th group. After calculating the F statistic it is used to calculate the p-value using a F-distribution with $r - 1, n_t - 1$ degrees of freedom. Using ANOVA allows one to evaluate the significance of overall results without the use of numerous t-tests while still being mathematically rigorous. This step of the two-stage process does not evaluate the significance of differences between individual groups however, requiring the second stage in the process.

$$SS_{\text{error}} = \sum_{i=1}^k \sum_{j=1}^{n_i} (Y_{ij} - \bar{Y}_i)^2 \quad (3.6)$$

$$F^* = \frac{MS_{\text{between}}}{MS_{\text{within}}} = \frac{SS_{\text{effect}}/(r-1)}{SS_{\text{error}}/n_t - r} \quad (3.7)$$

Violin Plots

A violin plot is a data visualization tool that combines aspects of box plots and kernel density plots to provide a comprehensive view of a dataset's distribution, central tendency, and variability [22]. The width of the plot represents the data density at different values, showing the distribution's shape, while the central line indicates the median. The violin plots are symmetrical and can display multiple distributions side by side for easy comparison. They are particularly useful for identifying data characteristics such as skewness, modality, and outliers, making them valuable for exploratory data analysis and comparing performance metrics across different groups or categories.

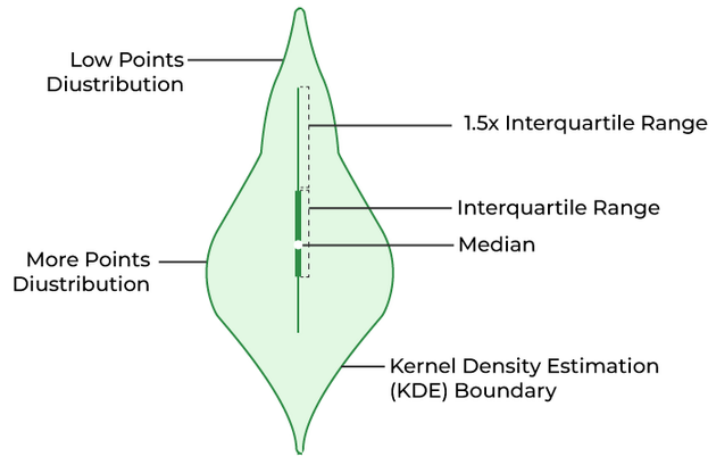


Figure 3.10: Violin plot Distribution Explanation [22]

A violin plot consists of four components:

1. **Median Point:** The white dot at the center represents the median of the distribution.
2. **Quartile Range:** The thin gray bar inside the plot shows the interquartile range

(IQR) of the data.

3. **Distribution Tails:** The long, thin lines extending outside the bar represent the distribution beyond the IQR. Points beyond these lines are considered outliers.
4. **Boundary Line:** The boundary of the violin plot is defined by a kernel density estimate (KDE) plot, illustrating the data distribution.

Chapter 4

METHODS

4.1 *Experiment Design*

The brain is always subject to constant stimulation from multiple sources; our senses capture all kinds of information, e.g., visual, auditory, tactile, olfactory. In addition, our brain is continuously orchestrating all our body's functions, from heartbeat to eye blinking. For these reasons, it is expected that EEG data (in the form of brain waves) may appear chaotic and noisy, and due to the nature of the data required for this research, it is necessary to set up experiments that allow us to focus on collecting relevant signals. Experiments need to focus on gathering brain waves that contain relevant information and that can be filtered from noise the best we can. This was the first challenge to consider for the design of our experiments, which consist of continuous sequences of activities that a subject must perform, e.g., analogous in some ways to a routine of PT exercises given to a patient. Such sequences can also be seen as “Game Flows”, or set of quests that a player would follow in order to receive scoring points.

On the other hand, a novel aspect of our work is the introduction and use of EMG, in the form of MyoBands as discussed in Chapter 3. This device allows us to capture signals coming from the muscles. The EEG and EMG devices are worn at the same time by a subject, and we aim to correlate signals from the muscles with those from the brain. A second challenge in here, is posed by the limitations of the EMG device itself, i.e., the kind of arm/hand gestures that can be detected by it. In addition, we needed to ensure that these actions also made sense in the context of our game, specifically, we want those actions that can be mapped into the game, e.g., navigation operation “move forward”.

The next challenge involved the identification of a limited set of arm/hand gestures picked

up by the EMG device, that could also generate large spikes in the brainwaves read by the EEG and that could be differentiated from other events (such as blinking) or isolated from the “noisy” signals that the body constantly generates.

In order to design our experiment sequences, we need to also take into account the scope of our research, which focuses on the study of upper limb conditions. For this, we derive ideas from tendon and nerve-gliding exercises to treat upper limb extremity weakness as mentioned in Section 3.4 and we define the set of actions that can be tracked by the EMG device, generate significant EEG changes and are relevant to our therapeutic goals. Figure 4.1 depicts these actions. These exercises are also commonly incorporated into the PT regimen to treat carpal tunnel syndrome and general upper limb weakness.

Definition 2. Let $\mathbb{A}$ represent the set of hand gestures that are detectable by our EEG and EMG devices, have therapeutic relevance (upper extremity), and can be mapped to a movement in the AR / MR environment.

Definition 3. In the context of this research, an ***action*** $A \in \mathbb{A}$, is a single instance of a hand gesture.

Definition 4. Let $\mathcal{T}$ represent a ***task***, which is an interval of time during which an action A_i is repeated several times.

Given that EEG will always be detecting changes in the brainwaves in a participant, we need to define a **baseline**. Having defined actions as intentional, detectable movements that a subject can do we can define a state of rest as our baseline:

Definition 5. Let $\mathcal{R}$ represent the state of **rest**, which is an interval of time during which no intentional actions are being made by the subject.

4.1.1 Game Flow Sequences

Definition 6. Let $\mathbb{G}$ represent a ***Game Flow***, which is a sequence of *tasks* each preceded and followed by resting intervals, i.e., $\mathcal{R}, \mathcal{T}_1, \mathcal{R}, \mathcal{T}_2, \mathcal{R}, \dots, \mathcal{R}, \mathcal{T}_n, \mathcal{R}$.

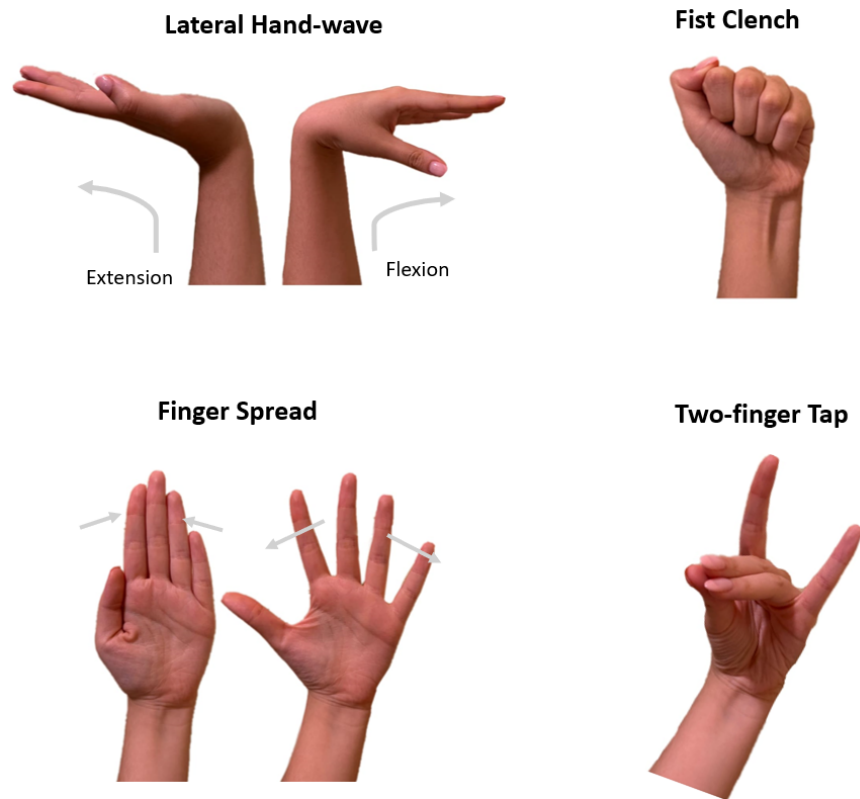


Figure 4.1: Hand Postures for experiments considered for this research. Lateral Hand-wave extension will informally be known as “Wave Out” and flexion as “Wave In”. Two-finger tap is also known as “Double Finger Tap” since it involves tapping twice with two fingers on the thumb.

The ***Game Flow*** will not only be used in this thesis but will also be important in future work, as discussed in chapter 6, section 6.2. Rest periods in EEG and EMG experiments are essential for establishing baseline measurements and reducing artifacts from fatigue. They also allow neural and muscle systems to recover, ensuring that data accurately reflects task-related responses. In the field of EEG and EMG research, it is common practice to use a 5-second duration for tasks followed by a 9-second rest period [3]. The tasks performed by the participants and their duration for this study are detailed in Table 4.1 ($\mathcal{R}$ is also included in this table for convenience).

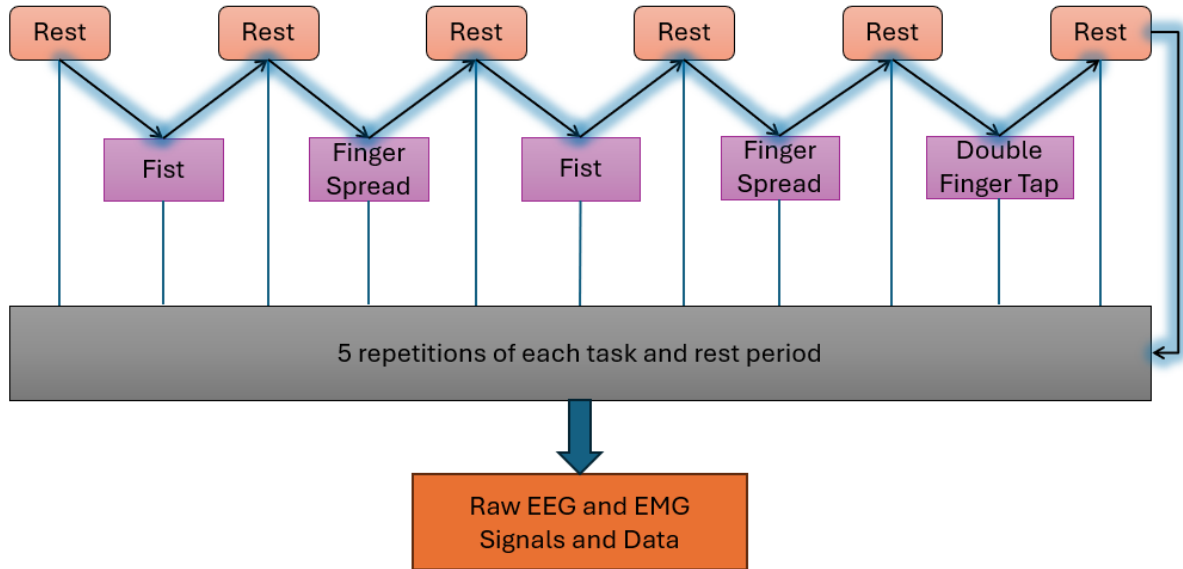
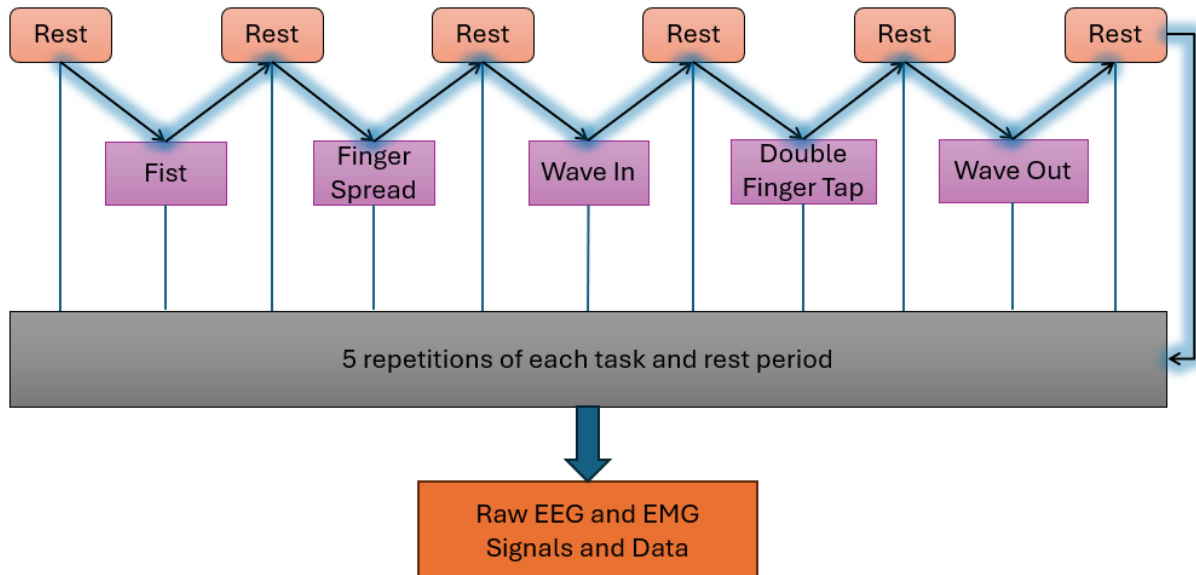
Table 4.1: Tasks considered for our experiments and their corresponding duration

Task $\mathcal{T}$	Duration (s)
Relax ($\mathcal{R}$)	10
Make a Fist	7
Spread Fingers	7
Hand Wave In (Flexion)	7
Hand Wave Out (Extension)	7
Two (Double) Finger Tap	7

In our experiments, the duration of rest $\mathcal{R}$ is 10 seconds, and the duration of any task $\mathcal{T}$ is 7 seconds. During those 7 seconds, the subject will perform a gesture A in a pulsating manner, for instance, during the 7 seconds of the Finger Spread task time interval, the subject will spread and collapse the fingers as shown in Figure 4.1, similarly, a Fist interval will consist of clenching and releasing the fist during 7 seconds.

We designed two game flows for this study. Figures 4.2 and 4.3 present flow diagrams with these Game Flows.

Definition 7. Let $\mathbf{S}$ be a ***session***, which is as a single iteration of some ***game flow*** $\mathbb{G}_i$ and lasts approximately 95 seconds.

Figure 4.2: Game Flow 1 (G_1) [1]Figure 4.3: Game Flow 2 (G_2) [1]

For a specific game flow $\mathbb{G}$, participants are instructed to follow tasks in order for the duration of the corresponding time, for instance:

$10(\mathcal{R}) + 7(\mathcal{T}_1) + 10(\mathcal{R}) + 7(\mathcal{T}_2) + 10(\mathcal{R}) + 7(\mathcal{T}_3) + 10(\mathcal{R}) + 7(\mathcal{T}_4) + 10(\mathcal{R}) + 7(\mathcal{T}_5) + 10(\mathcal{R})$
totals 95 seconds.

Each participant repeated a *session* 5 times back-to-back, and since there are two different *game flows*, each subject completed a total of 10 sessions. Hence, the overall amount of time a subject participated was:

$$(95 \times 5) \times 2 = 950 \text{ seconds} = 15 \text{ minutes and } 50 \text{ seconds.}$$

Definition 8. Let E_i represent an **experiment**, which is a set of $n \in [1, 5]$ back-to-back *sessions* of the *game flow* $\mathbb{G}_i$

In our work, each participant completed two experiments, one for each *game flow* and with $n = 5$. For the experiments, each participant is asked to sit in a comfortable position on a chair that is placed inside a room with as few distractions as possible. The EEG device is placed on their head and the EMG on their leading-hand forearm. As the experiment begins, the participant is verbally instructed to follow the tasks corresponding to a specific *game flow*.

To analyze and understand the data collected from each different task type, several methods of preliminary data analysis were performed, including plotting raw channel signals, frequency filtering, independent component analysis (ICA), event-related potential (ERP) and epoching. This will be discussed in detail in the following sections.

4.2 Data Collection

The data for this study was collected from volunteering UWB students and research team members. All participants provided informed consent prior to the start of the experiment.

Data was collected using two programs corresponding, respectively, to the EEG and EMG devices: the DSI-7 Streamer software from Wearable Sensing and Myo Connect software for MyoBands. The matMYO plug-in [12][11] was used for the session to collect data from the

Myoband. Raw EEG data was recorded and saved as an .edf (European Data Format) [8] file using DSI Streamer, while raw EMG data was saved as a .csv file.

EEG

The EEG data was recorded using a 7-channel WearableSensing DSI-7 dry EEG system with a sampling rate of 300 Hz. The electrode layout of the DSI-7 is shown in Figure 4.4 and has the following Channels:

- F3: Frontal left region.
- C3: Central left region.
- P3: Left posterior region.
- Pz: Central posterior region.
- P4: Right posterior region.
- C4: Central right region.
- F4: Frontal right region.

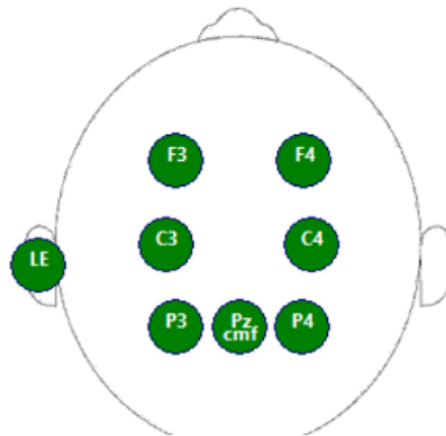


Figure 4.4: DSI-7 Electrode Layout. LE is a reference node that acts as a conglomerate of the other electrodes. F3 and F4 would correspond to the brain's Frontal Lobe, while the rest of the electrodes are on Parietal Lobe.

The DSI Streamer is used to connect to the EEG headset and collect data for each session. It is also used to calibrate each electrode and ensure a reliable connection. In addition, during data collection, the *event markers* for movements must be placed manually in the raw EEG file using DSI Streamer.

Figure 4.5 presents the high-level flow diagram for the EEG data collection.

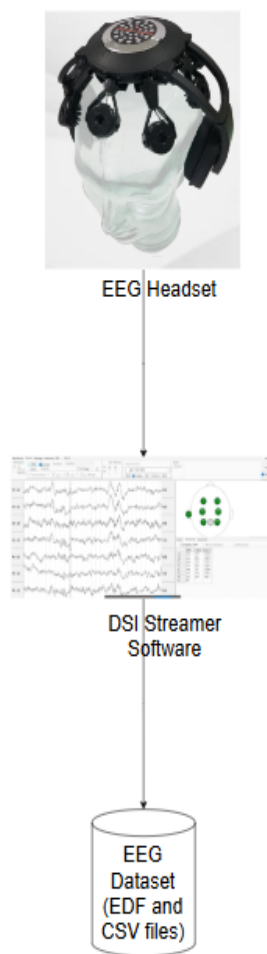


Figure 4.5: Data Collection Diagram [13][56]. Data is wirelessly transmitted from the EEG headset to DSI Streamer, which runs on a PC, where it is stored in the hard drive.

During our experimentation, we encountered various bugs in the DSI Streamer, some of

them documented in [13], and others found through our study. We have documented these findings in Appendix E together with panel images. Figure 4.6 displays the sample EEG data collected by DSI Streamer.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	# Mains_Fr	60													
2	# Sample_I	300													
3	# Filter_De	40													
4	# Sensor_C uV														
5	# Headset_DSI-7 SN:5122														
6	# Data_Log DSI-Streamer-v.1.08.60														
7	# Date = #####														
8	# Time = 15:51:08														
9	# Patient_ID XM01-JAN-0000 X														
10	# Record_ID Startdate 28-JUL-2024 EEGXXXXXXXXXXXXXXXXXX X X X														
11	# Filter = Non-Filtered														
12	# Comments =														
13	# Referenc Pz														
14	# Trigger S Wireless														
15	# Channel ch_1	ch_2	ch_3	ch_4	ch_5	ch_6	ch_7	Trigger	Time_Offs	ADC_Statu	ADC_Sequ	Event	Comments		
16	Time LE	F4	C4	P4	P3	C3	F3	Trigger	Time_Offs	ADC_Statu	ADC_Sequ	Event	Comments		
17	0.0033	-1251.6	-2179.2	-989.7	385.2	-38.7	861.3	-1532.1	0	6	0	132	1 Rest		
18	0.0067	-1128	-2058	-900.6	460.8	42.6	972.6	-1417.8	0	12	0	133	0		
19	0.01	-1107.6	-2033.7	-891.3	481.2	64.8	1002	-1389.9	0	18	0	134	0		
20	0.0133	-1209.9	-2134.5	-975	421.2	-6.3	916.5	-1487.1	0	24	0	135	0		
21	0.0167	-1309.2	-2231.1	-1062.9	352.8	-82.8	810.6	-1591.5	0	30	0	136	0		
22	0.02	-1256.4	-2165.7	-1000.2	395.1	-36.3	882	-1527	0	36	0	137	0		

Figure 4.6: Sample EEG Data in CSV format

EMG

The Myo Connect Software was used simultaneously to collect EMG data alongside EEG data from DSI Streamer. Initially, the MyoBand was connected to the Myo Connect Armband Manager, and a custom profile was set for each participant. For this configuration, the participants performed a series of five hand movements: making a fist, spreading fingers, waving in, waving out, and resting (See Figure 4.1). This process synced the MyoBands with their hands. Details about the Myo Connect Armband Manager can be found in the Appendix D.

After syncing the MyoBand, data collection is carried out using the matMYO plugin. First, the plugin is downloaded and opened in MATLAB. Then, the “Build_MYO_Mex”

MATLAB script is run, followed by the execution of the “myo_2c_sfunc” cpp script, which collects the data in a csv file format. This process gathers data from all eight EMG channels, together with IMU data (accelerometer and gyroscopic data on the x , y , and z axes). Figure 4.7 displays the sample EMG data collected by the matMYO plugin.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	timestamp	emg1	emg2	emg3	emg4	emg5	emg6	emg7	emg8	ax	ay	az	gx	gy	gz
2	1.72E+15	-2	0	-2	4	1	-1	-1	-1	0.073242	-0.77002	0.584961	-0.25	0.125	-0.25
3	1.72E+15	-2	0	-4	-8	-1	-1	0	0	0.073242	-0.77002	0.584961	-0.25	0.125	-0.25
4	1.72E+15	-3	-2	-3	-1	-1	0	-1	-1	0.073242	-0.77002	0.584961	-0.25	0.125	-0.25
5	1.72E+15	0	-1	-1	2	0	-2	-2	1	0.073242	-0.77002	0.584961	-0.25	0.125	-0.25
6	1.72E+15	1	-1	-2	-4	0	0	-1	-2	0.073242	-0.77002	0.584961	-0.25	0.125	-0.25
7	1.72E+15	0	0	0	0	1	0	1	1	0.073242	-0.77002	0.584961	-0.25	0.125	-0.25

Figure 4.7: Sample EMG and IMU Data in CSV format

4.3 Preprocessing Techniques

4.3.1 EEG Preprocessing

The raw EEG data collected during the motor task sessions was preprocessed using a combination of frequency filtering (FF), independent component analysis (ICA), and event-related potentials (ERPs).

Figure 4.8 below illustrates the data preprocessing pipeline for EEG data on EEGLAB used in the experiments.

The raw EEG data collected during the motor task sessions was preprocessed using EEGLAB. First, the raw EEG data was imported into EEGLAB. This initial step ensured that all data points were correctly integrated into the analysis environment. Event markers and channel locations were then imported too, providing necessary contextual information for the subsequent processing stages.

To enhance data quality, re-referencing and down-sampling were performed, and epoch baselines were removed. Band-pass frequency filtering, specifically within the 8–13 Hz range [49][17], was applied to isolate the frequencies of interest and minimize noise. The raw data was examined and the event labels were adjusted if necessary to ensure accuracy.

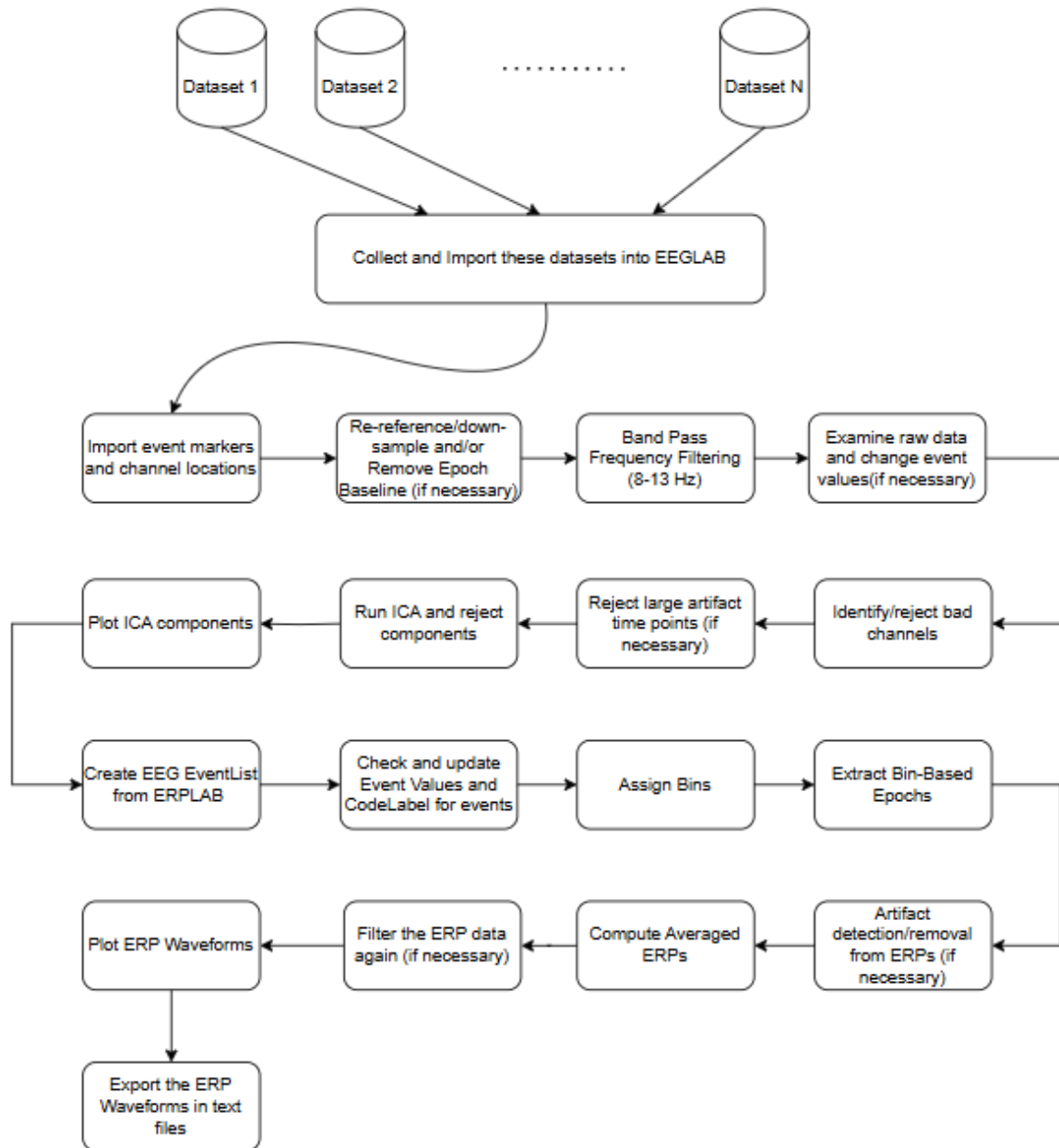


Figure 4.8: EEG Data Preprocessing Pipeline

The data collected from some channels was found to distort the overall data integrity, these channels were identified and rejected, if necessary. Additionally, large artifact time points were removed as required. Independent Component Analysis (ICA) was run to further clean the data, and components were rejected on the basis of necessity. The ICA components were plotted for visual inspection.

An EEG EventList was created from ERPLAB, and event values and CodeLabels were checked and updated for the tasks done during the experiments. Bins were assigned to categorize the data effectively, and bin-based epochs were extracted for detailed analysis.

Artifact detection and removal were performed on the ERPs if necessary. Averaged ERPs were computed to facilitate a clearer understanding of the underlying neural activity. If required, the ERP data was filtered again to refine the results further.

Finally, ERP waveforms and scalp maps were plotted to visualize the data comprehensively. The processed ERP waveforms were then exported as text files for further analysis and documentation. This comprehensive preprocessing pipeline ensured the EEG data was clean, accurate, and ready for subsequent analysis stages.

4.3.2 EMG Preprocessing

The raw EMG data collected during the motor task sessions using the matMYO plug-in underwent preprocessing through a combination of techniques, including frequency filtering (FF), feature extraction, and segmentation. Figure 4.9 below shows the complete form of the data preprocessing of EMG data pipeline on MATLAB for the experiments.

The raw EMG data was collected and stored in .csv files using the matMYO plug-in. The data was saved into different variables for EMG and IMU (accelerometer and gyroscopic) data. If necessary, the data was synchronized and resampled to ensure consistency.

To preprocess the data, filters were applied to both EMG and IMU signals. A band-pass filter between 20 and 450 Hz was used for EMG data, while a low-pass filter below 20 Hz was applied to IMU data [31]. The original and filtered data were plotted to visualize the filtering effects.

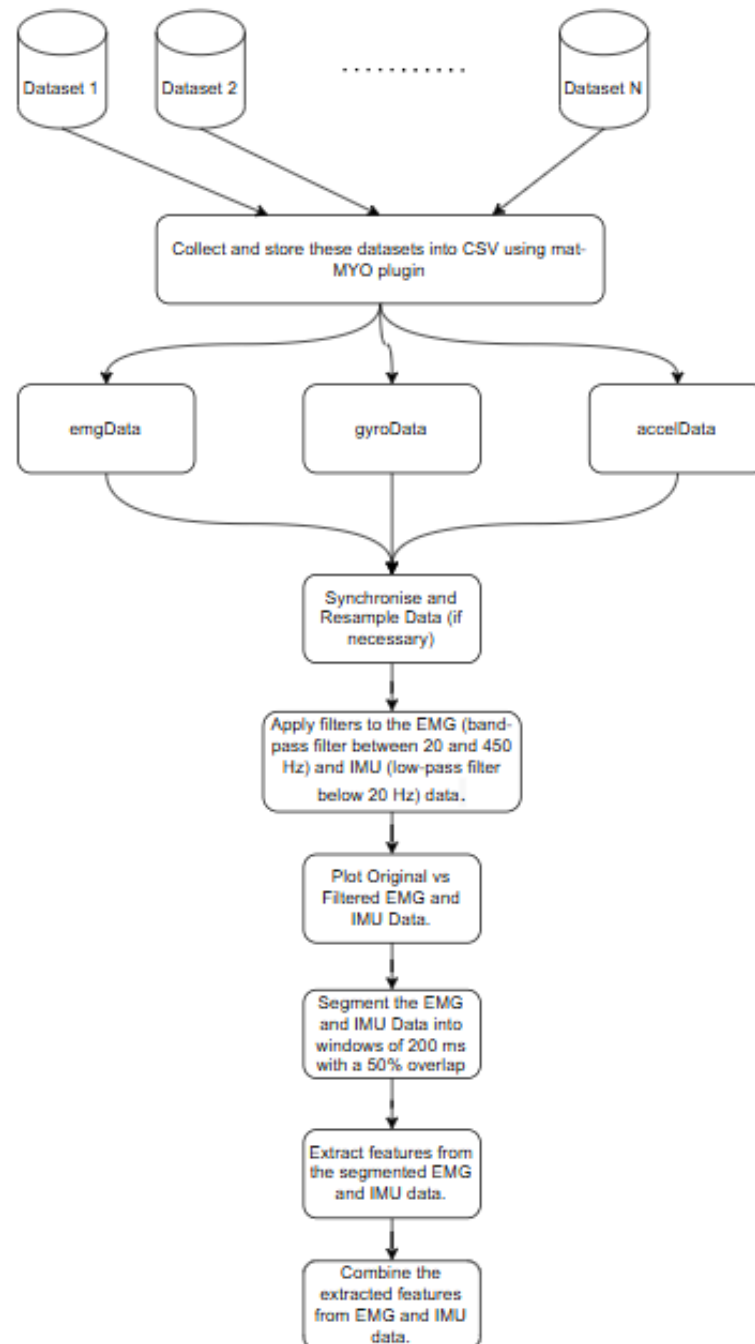


Figure 4.9: EMG Data Preprocessing Pipeline

The data was then segmented into 200 ms windows with a 50% overlap to capture detailed information about muscle activity and movement. Features were extracted from the segmented data, including mean absolute value, root mean square, variance, mean, standard deviation, and more.

Finally, the extracted features from both EMG and IMU data were combined for further analysis and classification. These features include Mean Absolute Value (MAV), Root Mean Square (RMS), Variance, Zero Crossing Rate (ZCR), and Waveform Length (WL) for EMG data and Mean and Standard Deviation for the IMU data. There are 5 distinct EMG features, and EMG data has 8 channels, the total number of EMG features is calculated as follows: $(5 \text{ EMG features} \times 8 \text{ EMG channels}) = 40 \text{ features}$. There are 2 features for the IMU data (the accelerometer and the gyroscope). Each of these components has 3 axes (x, y, z). Therefore, the total number of axes is 6. Since each axis has 2 features, the total number of features for the IMU data is calculated as follows: $(2 \text{ IMU features} \times 6(3 \text{ accelerometer axes} + 3 \text{ gyroscope axes})) = 12 \text{ features}$. Hence, the total number of EMG-IMU features is $40 + 12 = 52 \text{ features}$.

4.4 3D Simulation Design

In the 3D Simulation, the process begins with the Intro Scene, where users select their character using hand gestures with the Myo Armband. After choosing a character and the flow of the game, users start the gameplay with a fist gesture. During the game simulation, users interact through predefined hand gestures associated with the selected game flow. The game flow data is utilized to train machine learning (ML) models and develop a prediction module after processing EEG and EMG data. The prediction models then interface with the feedback application, which recommends the next-hand movement for the game. Figure 4.10 shows the flow architecture of the game.

Additionally, a free game flow scene allows users to create their own game flows, though this feature is beyond the scope of the current thesis and is suggested for future development in chapter 6, section 6.2. The feedback and recommendation system is designed to enhance

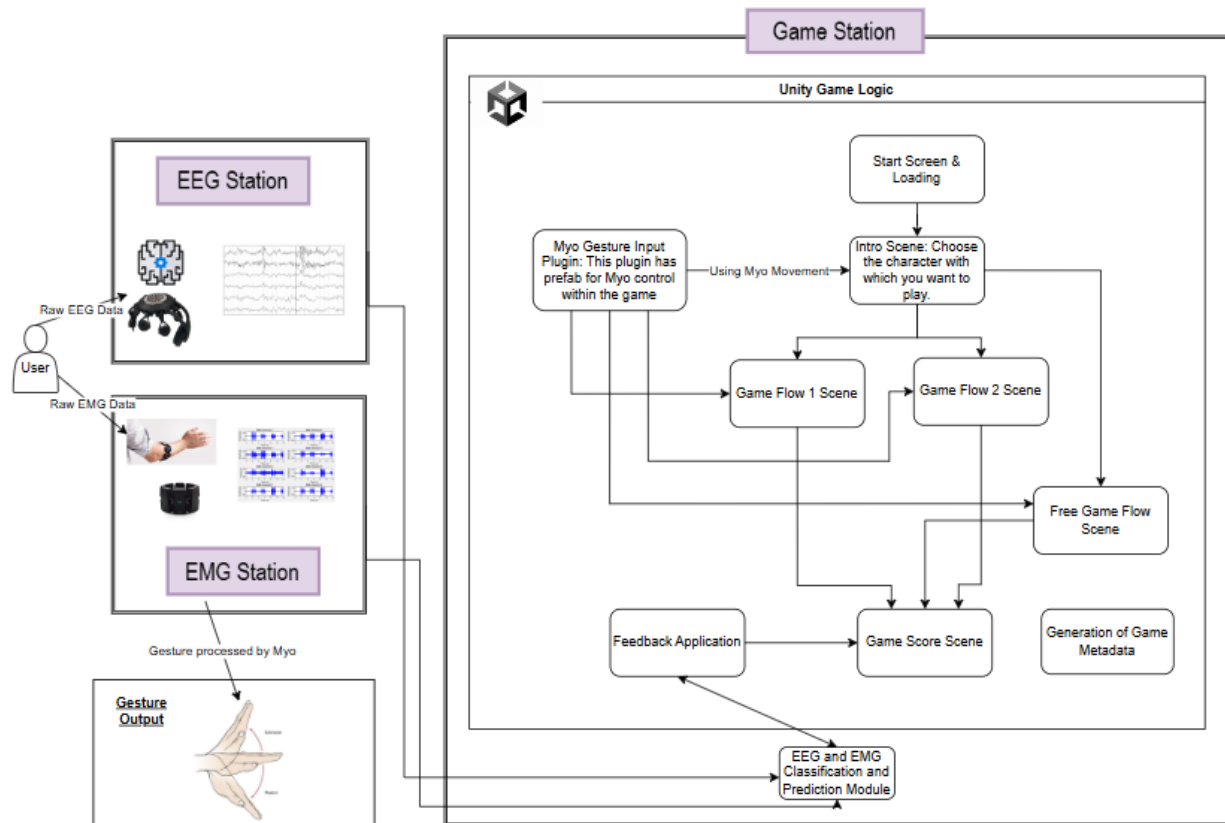


Figure 4.10: Game Architecture

user experience by utilizing detailed metrics and features derived from the classification models of EEG and EMG data. Key metrics include Gesture Recognition Accuracy, which measures how accurately the system identifies and classifies hand movements, and the timing of hand movements, which tracks the duration and sequence of gestures based on the game flows.

Overall, this approach ensures that feedback and recommendations are closely aligned with the user's actual performance and needs, leading to more effective and personalized training and rehabilitation experiences.

4.5 Classification Models and Statistical Analysis

To classify the preprocessed EEG and EMG data, several different models were implemented. These machine learning models are as follows: Support Vector Machine (SVM), Convolutional Neural Network (CNN), Feedforward Neural Network (FNN), Recurrent Neural Network (RNN), and Long Short-Term Memory (LSTM). Figure 4.11 illustrates the classification and analysis done on the EEG data. Figure 4.12 shows the classification and analysis done on the EMG data. Although the classification used features from both EEG and EMG data in a multimodal fusion, the figures present the processes separately for clarity. For EEG and ERP analysis, the extracted features were mean, standard deviation, minimum value, maximum value, range, median, skewness, kurtosis, peak value P300 and peak time P300 of the ERP data. There are 10 distinct EEG-ERP features, and with 7 EEG channels, the total number of EEG-ERP features is calculated as follows: $(10 \text{ EEG-ERP features} \times 7 \text{ EEG channels}) = 70 \text{ features}$.

Support Vector Machine (SVM) is a supervised learning algorithm used for classification and regression. It finds the optimal hyperplane that maximizes the margin between classes, making it effective in high-dimensional spaces and robust to overfitting.

Convolutional Neural Network (CNN) is a deep learning algorithm effective for image and spatial data processing. It uses convolutional layers to learn spatial hierarchies of features, making it suitable for tasks such as image classification and object detection.

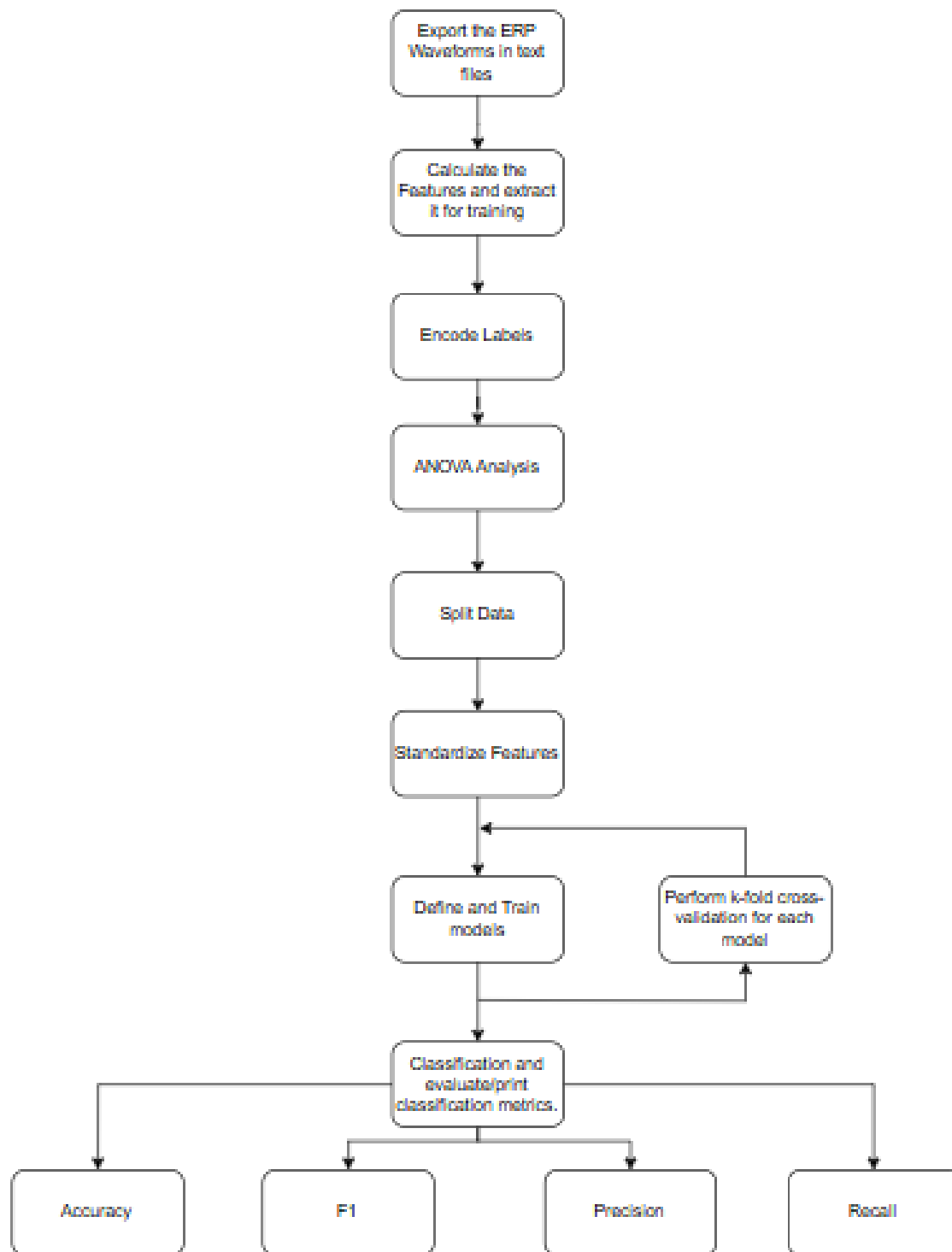


Figure 4.11: EEG data Classification and Statistical Analysis Layout

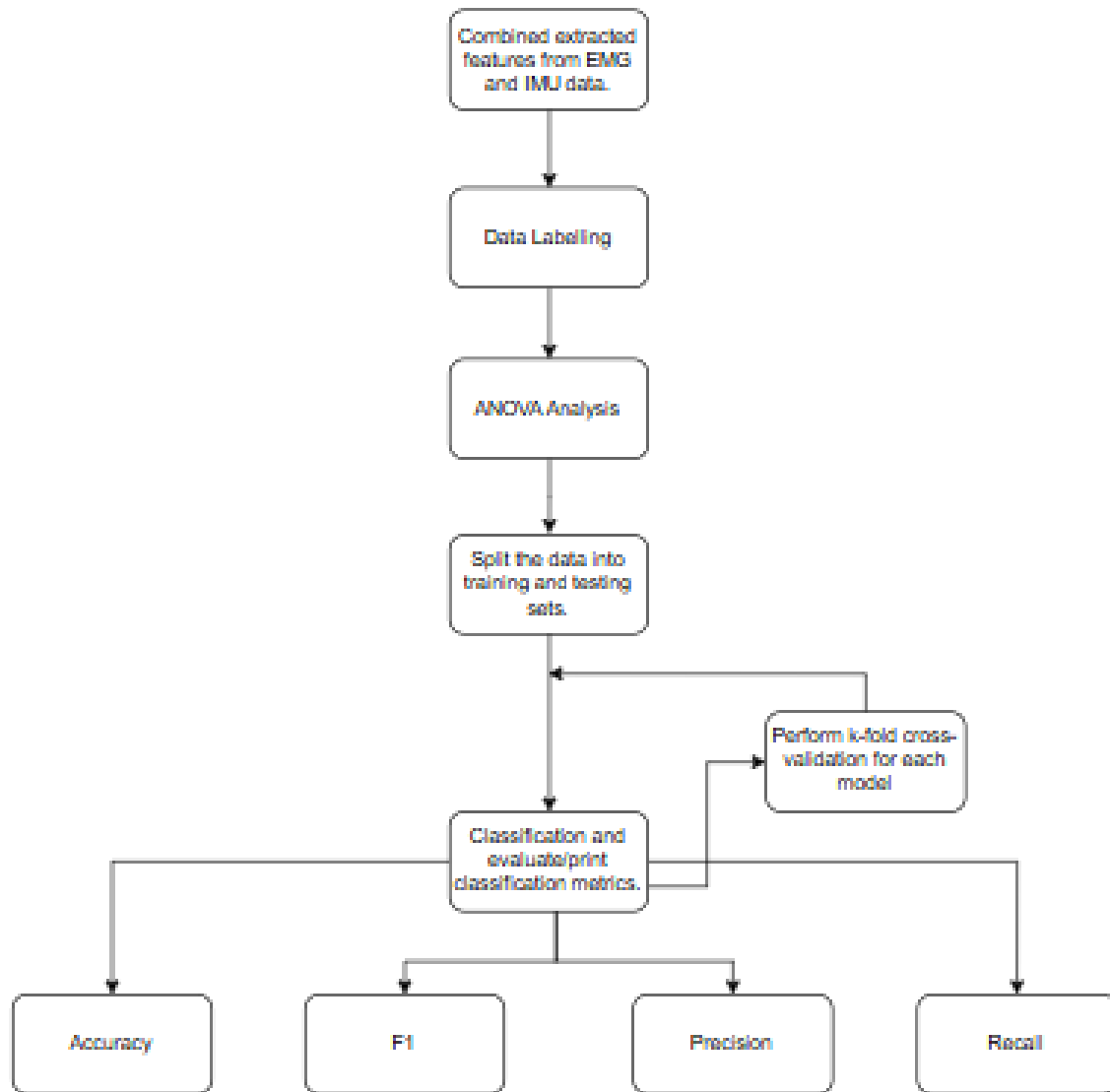


Figure 4.12: EMG data Classification and Statistical Analysis Layout

Feedforward Neural Network (FNN) is an artificial neural network with a straightforward architecture of input, hidden, and output layers, where information moves in one direction. It is used for various classification and regression tasks and serves as a basic building block for more complex neural networks.

Recurrent Neural Network (RNN) is designed for sequential data processing, using feedback loops to maintain memory of previous input. This makes it suitable for tasks like time-series analysis and natural language processing, although it can struggle with long-term dependencies.

Long-Short-Term Memory (LSTM) is a type of RNN that handles long-term dependencies through a set of gates that regulate information flow. This makes LSTMs well-suited for tasks requiring understanding of long-term sequences, such as speech recognition and time-series forecasting.

The performance of each model was evaluated on the basis of accuracy, precision, recall, and F1 score metrics.

Overall, these classification models were selected to compare their performance and identify the most accurate one to classify motor tasks based on the preprocessed EEG and EMG data.

4.5.1 Statistical Analysis

To evaluate the statistical power of the classified data, analysis of variance (ANOVA) was utilized and is defined in Section 3.5. ANOVA was used to assess the overall power of the data and to analyze the effects of each independent variable on the dependent variable. Details for each of these methods can be found in chapter 3, section 3.5.

The ANOVA test was used to see which feature sets worked better for classification and to check how significant the differences were, using p-values and F-values. If differences were significant, we looked closer to find out which specific pairs of features were different.

Statistical analysis was performed using the SkLearn, Seaborn, and Scipy statistical software packages of Python.

4.6 Evaluation Metrics

To evaluate the effectiveness of the classification models, we will use several performance metrics, including accuracy, F1 score, precision, and recall.

Accuracy: The most commonly used metric that measures the overall correctness of the classification results. It is calculated by dividing the number of correct predictions by the total number of predictions.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$$

Recall: It measures the proportion of true positive predictions among all the actual positive instances in the dataset. High recall indicates that the model is making fewer false negative predictions.

$$Recall = \frac{TP}{TP+FN}$$

Precision: It measures the proportion of true positive predictions among all the positive predictions made by the model. High precision indicates that the model is making fewer false positive predictions.

$$Precision = \frac{TP}{TP+FP}$$

F1 Score: It is a measure of the balance between precision and recall, calculated as the harmonic mean of the two. It provides a single score that considers both precision and recall, which is useful when the classes are imbalanced.

$$F1\ Score = 2 \times \frac{precision \times recall}{precision + recall}$$

We will also use confusion matrices to visualize the classification results and to compute other evaluation metrics such as specificity and sensitivity. The evaluation metrics will help us determine the best classification model for the EEG and EMG data classification task.

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Figure 4.13: Confusion Matrix showing relationship between actual and predicted values in the form of true or false positive and negatives [21].

4.7 Software and Tools

For data collection, we used DSI Streamer [56] from Wearable Sensing, Myo Connect Software, and matMYO[12][11] plugin to collect data from student volunteers. These software tools were installed on a laptop and used to collect EEG and EMG data from multiple channels simultaneously.

Data preprocessing for EEG signals was conducted using EEGLABv2024.0 [7], an open-source MATLAB toolbox for EEG data analysis. The .edf file from DSI Streamer was imported into EEGLAB with the BIOSIG plugin. Details about EEGLAB's interface and plugins are provided in the Appendix A. EEGLAB used the following plugins[7] for EEG data preprocessing: BIOSIG, ERPLAB, ICLabel, WearableSensing, erppeakinterval, erp-source. Data preprocessing for EMG data was performed using the matMYO plugin in

MATLAB R2023b, while classification models were developed with Python 3.12.4, utilizing libraries such as Pandas [52], NumPy[18], scikit-learn[45], TensorFlow[51], Seaborn[55], Matplotlib[25], and SciPy[19].

The classification models were executed in PyCharm 2024.1.3 [28] using the Jupyter package available for PyCharm. All models were run in Python for processing both EEG and EMG data.

For game/3D simulation development, Unity Hub 3.8.0 and Unity 2022.3.17f1 [54] were used. The Unity packages utilized include Terrain Tools, AR Foundation, Apple ARKit XR Plugin, Google ARCore XR Plugin, FBX Exporter, and Animation Rigging. The characters and animations were sourced from Mixamo[37] .

Overall, these software and tools provided a robust and efficient framework for data analysis, preprocessing, game/simulation development, and classification model implementation, enabling us to achieve the objectives of this research.

Chapter 5

RESULTS

In this chapter, we discuss in detail the data collected and results based on the methods described in Chapter 4.

5.1 Data Collection Results

Over the course of the experiment the data collection process involved 5 volunteers who each participated in a data collection session for a motor task using electroencephalography (EEG) and Electromyography (EMG) equipment over several days.

A total of 23 *sessions* were conducted for each *game flow*, resulting in a total of 46 data collection sessions. Data collection was carried out in a quiet room with minimal distractions to ensure the accuracy and quality of the recordings. Participants wore an EEG device on their head and an EMG device on their leading-hand forearm. No additional data was recorded regarding the participants' physical or medical history, nor their preferences for tasks. Furthermore, no explicit events were logged outside of the structured timing for each session, which required the motor tasks to start and stop according to the predetermined schedule.

The datasets we collected gave us plenty of information to look at. We used these data to test different ways to prepare the data and various machine learning models for classifying motor tasks. The results of this work are shown in the following sections.

5.2 Raw EEG and EMG Plot Analysis

After collecting the data, the first step was to examine the raw EEG and EMG signals from the motor tasks. This visual inspection was designed to find patterns and event markers that

corresponded to the motor tasks. Since the motor cortex and muscle movements generate strong signals, these were expected to stand out in the data. Figures 5.1 and 5.2 below illustrates examples of how brain activity and muscle movements change throughout a data collection session. Each task showed different patterns that disturbed the normal resting state of both the brain and muscles, based on the cues for the motor task. Machine learning models were believed to be able to successfully tell the difference between positive and negative signals if the data were divided into epochs. In this case, *positive* means that there is a noticeable fluctuation of the signal in the presence of a motor task.

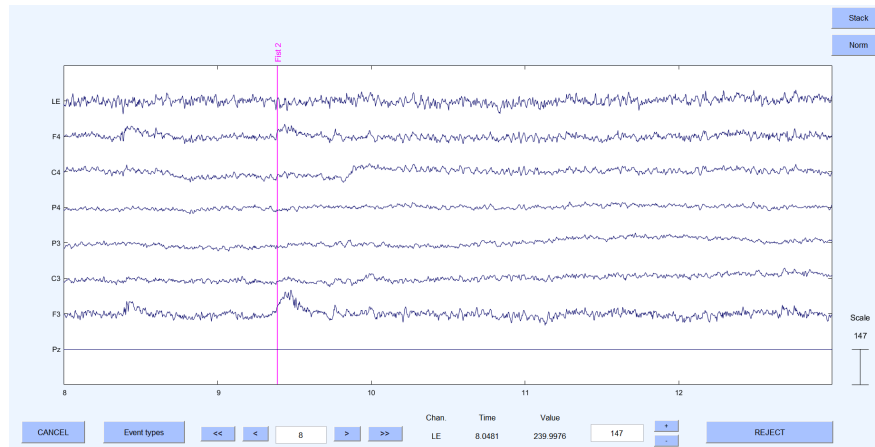


Figure 5.1: Plots of each Channel of Raw EEG Data collected for Game Flow 1 during one session using Matlab.

In general, the EEG and EMG waveforms showed considerable variability between channels, with each channel exhibiting distinct behavior. This initial analysis of the raw EEG and EMG data laid the foundation for further investigation, shaping the direction of the research, and guiding subsequent analysis and findings.

Figures 5.3 and Figure 5.4 shows 4D scatter plots of accelerometer and gyroscopic data respectively collected using the Myoband. The plot shows the acceleration on three axes (X, Y, Z), with a color bar indicating a time of 0 and 95 s (length of the *session*).

Upon visual inspection, we can see clustering patterns in the accelerometer data asso-

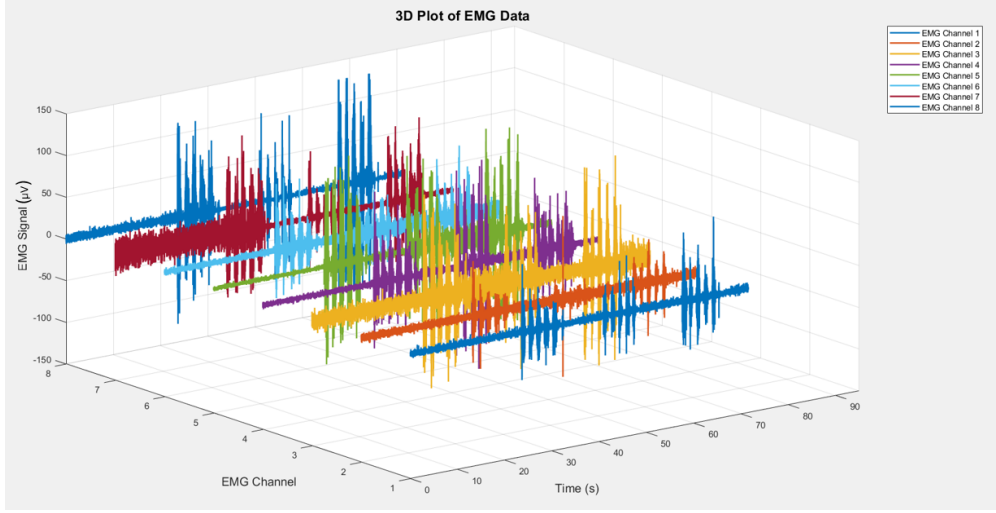


Figure 5.2: Each trend represents a each Channel of Raw EMG Data for one session of $\mathcal{G}_1$. Note how the signal shows disruptions compared to the baseline ($\mathcal{R}$) wave that starts at time 0.

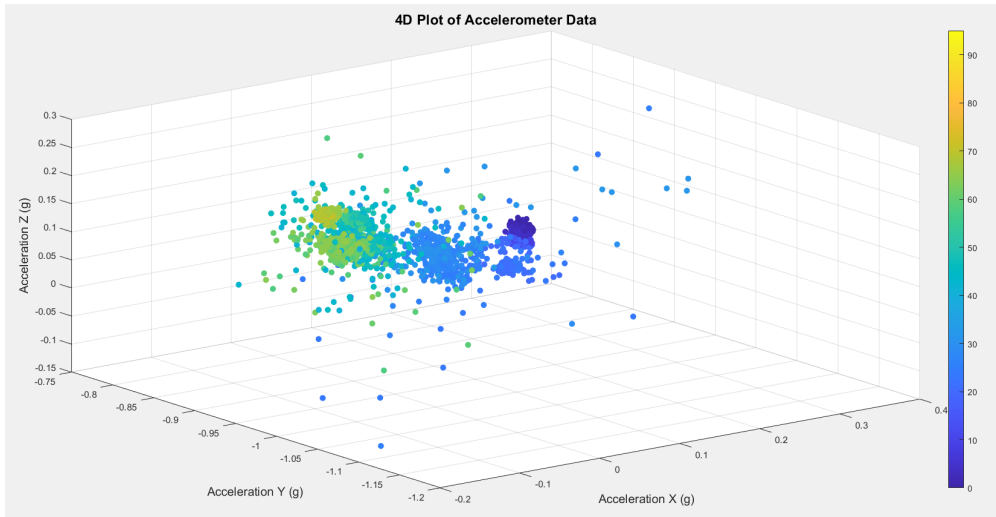


Figure 5.3: Each axis of Raw IMU Accelerometer Data corresponds one session of $\mathcal{G}_1$ and time is captured in a heat-map fashion using the bar on the right.

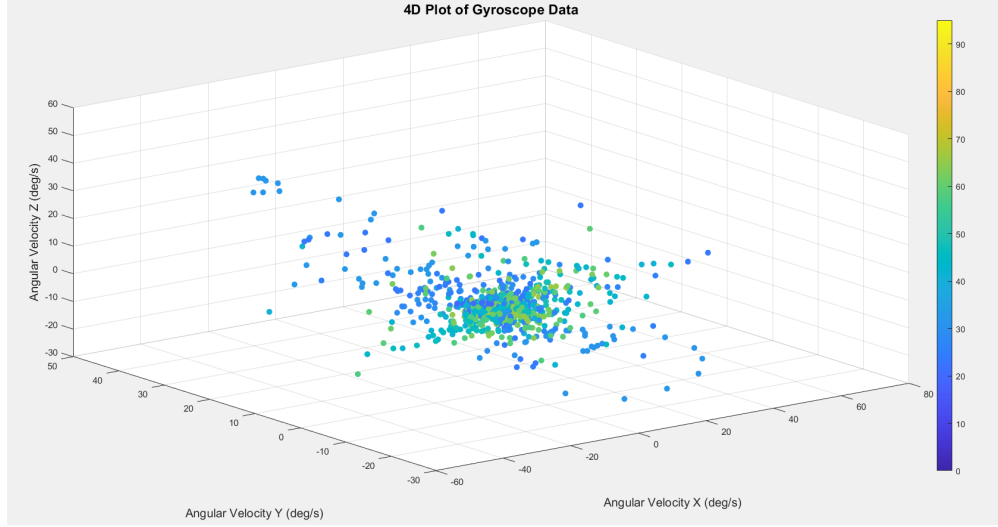


Figure 5.4: 4D Plots of each axis of Raw IMU Gyroscopic Data for one session of Game Flow 1.

ciated with the time, however, while these clusters are harder to discern in the gyroscope data, the pattern across time seems to go from the edges to the center of the data cluster. A prominent cluster at (0,0,0) in the accelerometer plot indicates the rest position of the MyoBand, which was maintained for 60 seconds out of the total 95 seconds. Both of these patterns can be explained based on the fact that during a session, ordering the different tasks of the *game flow* would cause changes detectable by these sensors.

5.3 Preprocessing Results

The preprocessing results are presented for one volunteer's one session during Game Flow 1. Therefore, all results belong to a single session of Game Flow 1 for the Pre-processing Results section.

5.3.1 EEG Preprocessing Results

The preprocessing techniques applied in this study comprised frequency filtering (FF), independent component analysis (ICA), event-related potentials (ERP), epoch, segmentation, and feature extraction. Their effectiveness was assessed by examining their impact on classification metrics—including accuracy, F1 score, recall, and precision—across all the machine learning models used. We used EEGLAB[7] for EEG preprocessing.

Channel Location In EEGLAB, channel location refers to the spatial coordinates of the electrodes used in EEG signals. These locations are crucial for accurate interpretation of the data, as they define where on the scalp each electrode is placed. Channel locations are typically represented in terms of Cartesian coordinates (x, y, z) or as spherical coordinates (theta, phi) relative to a common reference point, such as the center of the head.

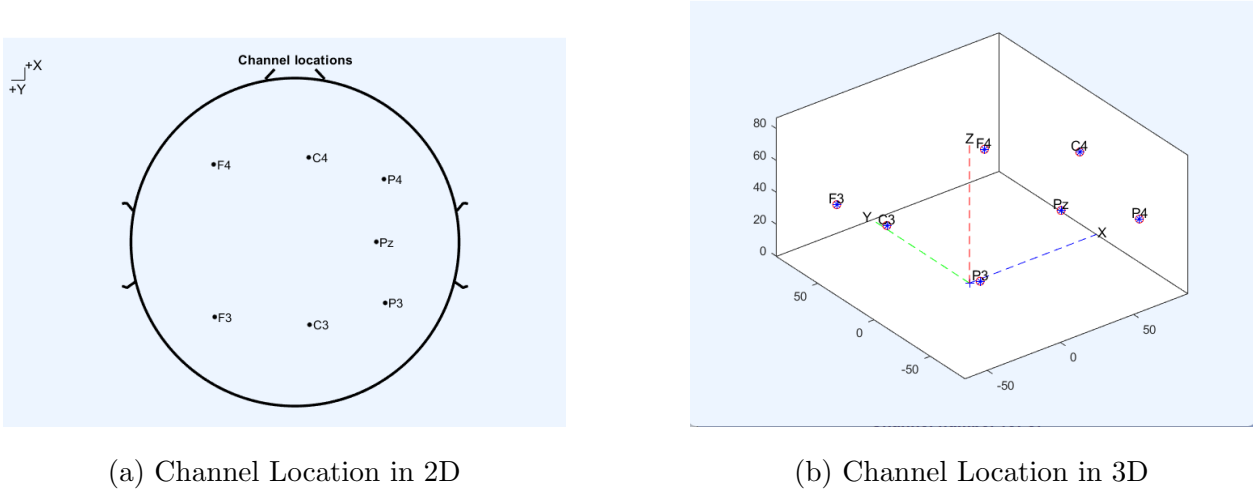


Figure 5.5: Channel Locations in 2D and 3D

A key step in analyzing EEG data is using frequency filters to reduce noise. Band-pass filters, when set to the right wavelength, can help identify certain actions by highlighting changes in brain waves. In this study, we used a bandpass filter between 8-13 Hz. The

figure 5.6 shows the results, with graphs of frequency vs magnitude (dB) and frequency vs. phase (degrees).

Magnitude: In signal processing, magnitude refers to the strength of a signal at different frequencies. When a filter is applied, the magnitude response shows how much of each frequency component passes through, measured in decibels (dB). Higher magnitude means less attenuation, while lower magnitude indicates more attenuation.

Phase: Phase represents the time shift of a signal's frequency components. When a signal passes through a filter, different frequencies may be delayed differently, as shown in the phase response, measured in degrees. A linear phase response is ideal because it ensures all frequency components are delayed equally, preserving the waveform shape.

Figure 5.6 shows the magnitude and phase responses of a band pass filter with a range of 8-13 Hz. The magnitude response graph shows that the filter successfully isolates the 8-13 Hz frequency band by reducing the strength of frequencies outside this range. Within the 8-13 Hz band, the response is flat, meaning these frequencies pass through with little reduction. This is important for analyzing EEG signals in the alpha band.

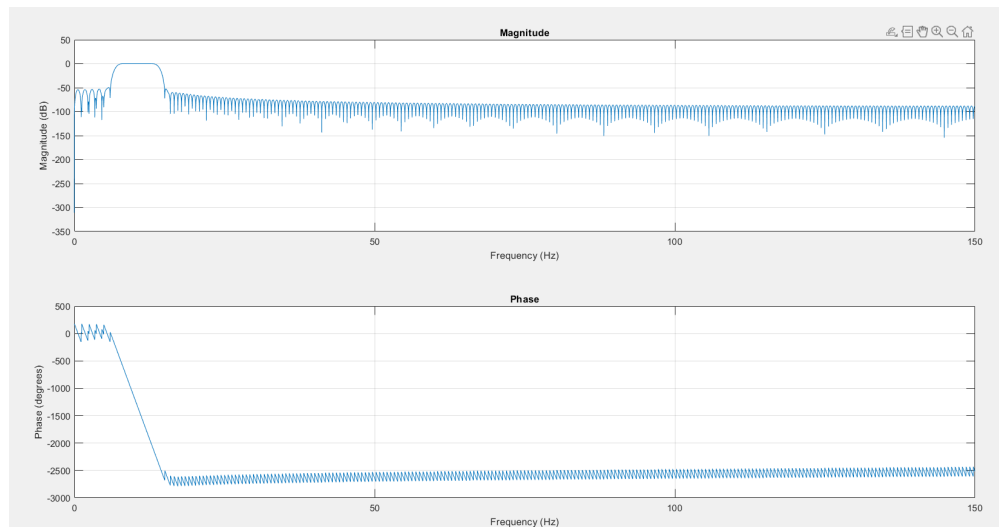


Figure 5.6: Plot of phase and magnitude vs. frequency after frequency filtering using a band pass filter of 8 – 13 Hz for a session in $\mathcal{G}_1$.

The phase response graph shows a sharp initial decline followed by a gradual descent. This means that the filter causes a phase shift, with different frequencies experiencing different delays. Understanding this shift is important for interpreting the timing of the signal components, as it affects the overall shape of the filtered signal.

Figure 5.6 displays the magnitude and phase plots of the frequency components of a signal, generated using the Fast Fourier Transform (FFT). The FFT is a mathematical algorithm that transforms a time-domain signal into its frequency-domain representation, breaking it down into its constituent frequencies.

Overall, the filter effectively isolates the 8-13 Hz frequency band while maintaining a manageable phase shift, making it well suited for EEG signal analysis.

Next, we applied Independent Component Analysis (ICA). Figure 5.7 shows the components resulting from this process. It displays the properties of seven components, each represented by a topographical map indicating brain activity levels. The maps are labeled with percentages ranging from 90.6% to 100.0%, suggesting the precision or intensity of the brain activity detected in each component. Components 1, 2, 3, 5, 6, and 7 show high activity levels above 99%, while component 4 shows a slightly lower activity level at 90.6%. Higher percentages indicate that the components are likely to capture significant and relevant brain activity, making them valuable for further analysis.

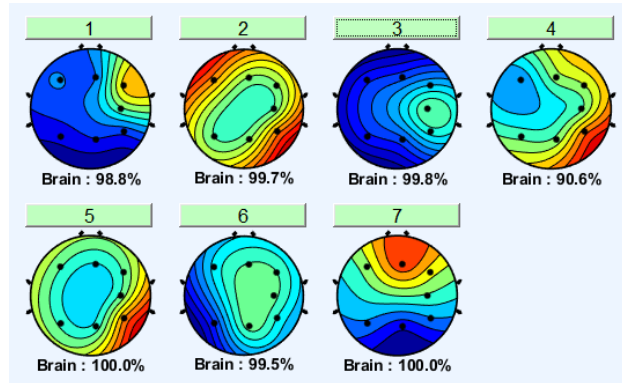


Figure 5.7: ICA IC Label components for a session of $\mathbb{G}_1$

Using the EEGLab tool we can dive deeper into each of the components (see Figure 5.8).

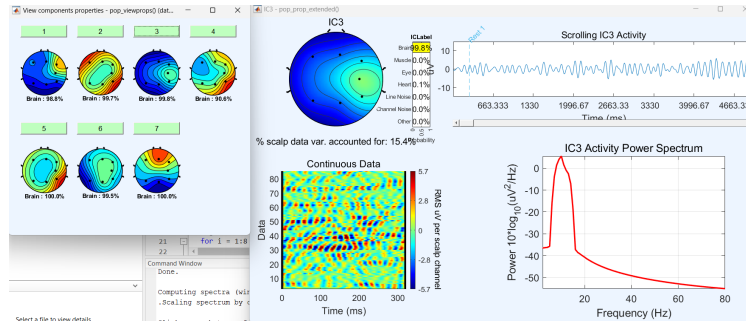


Figure 5.8: Selecting IC3 for a session of $\mathbb{G}_1$

Figure 5.9 shows a compound figure containing different graphs detailing the information for each component. In this case, we selected IC3 based on 5.7. IC3 is in dark blue in color indicating less brain activity in this IC component. The figure 5.10 shows IC5 from the same session's analysis. You can see that IC5 is more active than IC3. Here are the components we looked at in these figures:

IC3 and IC5 Component Variability Map: This chart shows that IC3 accounts for 99.1% of the variability in brain activity within the dataset, highlighting its significant role in the analysis. It is a key component and a major factor in the dataset. On the other hand, IC5 shows 100% of brain activity, indicating that it is highly active and influential as well.

Scrolling IC3 and IC5 Activity: The line graph illustrates how the IC3 signal varies over time, showing distinct peaks and dips. These changes indicate fluctuations in brain activity and help us see how the signal evolves. For IC5, there are noticeable differences after the event markers during the scrolling activity, suggesting brain activity.

Continuous Data Heatmap: The heatmap shows different patterns of activity across electrode sites over time. The red and orange areas indicate high activity periods, suggesting bursts of brain activity at specific moments. Some electrodes consistently show more activity, pointing to specific brain regions. The heatmap also reveals alternating high and low activity

periods, showing how brain activity changes with different tasks. This helps us understand which brain regions are more active at certain times.

IC3 and IC5 Activity Power Spectrum: The power spectrum graph for the IC3 and IC5 component shows that the signal has significant power in the low-frequency range, particularly around 8-13 Hz, which corresponds to the alpha band in EEG signals. This dominance indicates that the brain activity captured by IC3 and IC5 oscillates primarily at these frequencies, which are associated with relaxed, awake states. As the frequency increases, the power decreases, a common characteristic of brain signals where higher frequencies have lower power.

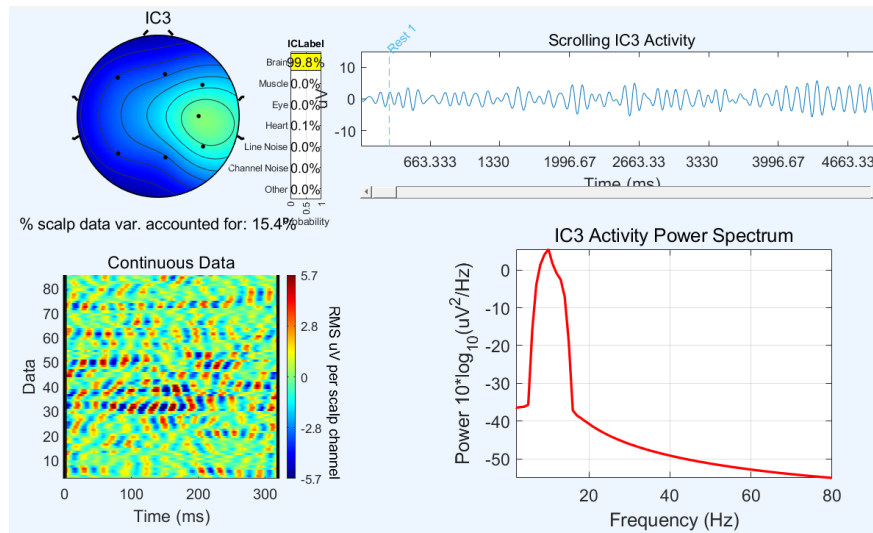


Figure 5.9: Detail of IC3 for a session of G_1 . Top Left: Variability summary for the component IC3. Top Right: ICA Activity timeline. Bottom Left: Data Heatmap. Bottom Right: Activity Power Spectrum.

Next, we will examine the same ICA components in 3D. Figures 5.11 and 5.12 illustrate these components in three dimensions.

The 3D thermal models of the heads show different patterns of brain activity, with variations in color indicating different levels of activation. Warmer regions (red) suggest higher

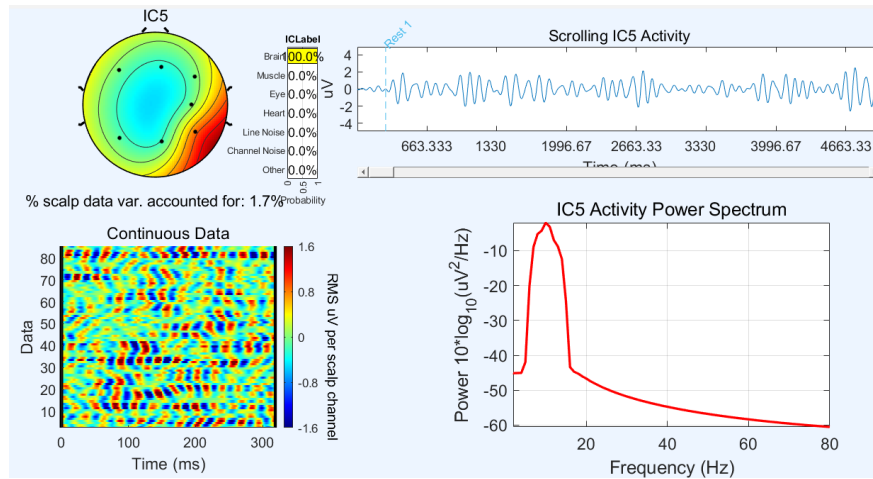


Figure 5.10: Detail of IC5 for a session of $\mathbb{G}_1$. Top Left: Variability summary for the component IC5. Top Right: ICA Activity timeline. Bottom Left: Data Heatmap. Bottom Right: Activity Power Spectrum.

activity, while cooler regions (blue) indicate lower activity. Each head represents a different independent component, highlighting how brain activity is distributed across various components.

After that, we will review the results from the bin-based epoched data for Artifact Detection and/or Rejection, used to create the ERPs, as shown in the figure 5.13.

The waveforms show clear peaks and troughs at similar times across different channels, indicating event-related potentials (ERPs) linked to specific sensory, cognitive, or motor tasks. The vertical lines across the panels mark key moments related to the task, showing when tasks were performed or when participants responded. Figure 5.13 shows that "B1_Rest" exhibits lower EEG signal activity, while "B3.FingerSpread" displays increased EEG signal activity. Labels like "B6_DoubleTap" and "B5(FingerSpread)" suggest that the data are organized into bins based on different experimental conditions or tasks performed during the EEG recording.

The consistent patterns across multiple channels suggest that the data is generally clean

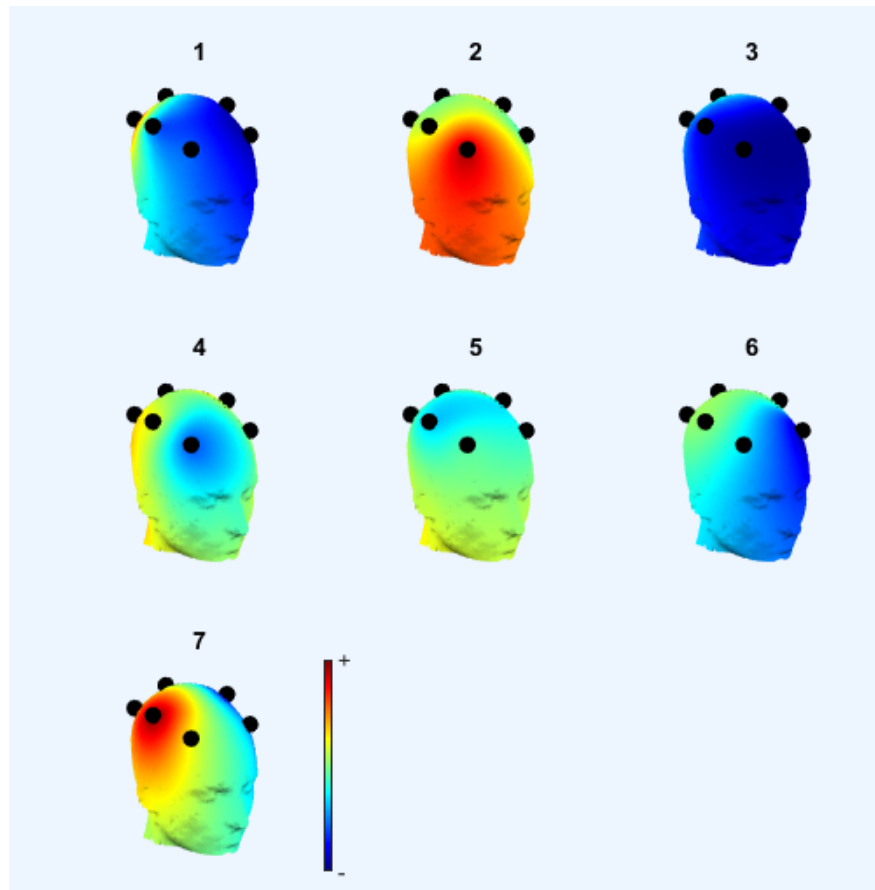


Figure 5.11: ICA Component Maps 3D for one session in Game Flow 1

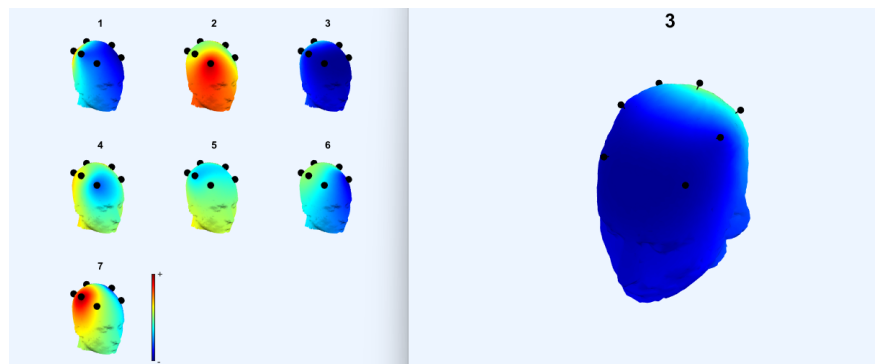


Figure 5.12: ICA - IC3 Component Maps in 3D for one session in Game Flow 1

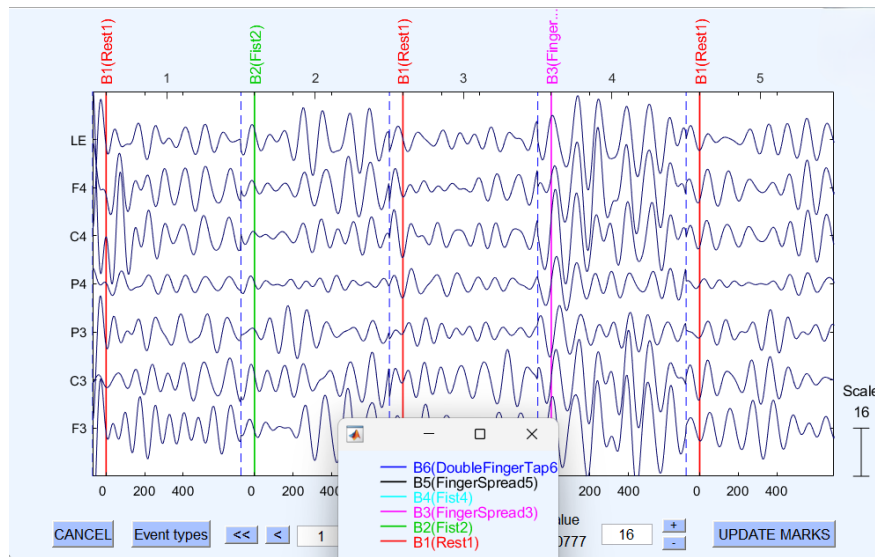


Figure 5.13: Artifact Detection in Bin Based Epoched Data for one subject's session for Game Flow 1

with few artifacts. Any irregularities were addressed through artifact rejection to ensure accuracy. Overall, the figure illustrates the electrical activity of the brain in response to different stimuli, highlighting the importance of detecting and removing artifacts for reliable ERP results.

The ERP set in the figure 5.14 shows brain responses recorded from all channels and tasks. Each graph shows multiple waveforms in different colors for each task. The x-axis represents time from -70 to 700 milliseconds, and the y-axis shows voltage in microvolts. The waveforms have clear peaks and troughs, showing how the brain reacts to different motor tasks. The Fist and Finger Spread tasks show bigger changes compared to the Rest condition, indicating more brain activity during these tasks. Consistent patterns across channels reveal how the brain coordinates responses to movement, offering insights into how movement is processed in the brain. Figure 5.15 illustrates the ERP for channel F4, which corresponds to the right hand during task performance. The F4 channel, located in the frontal lobe and is involved in motor control, is known to be more active during hand movements..

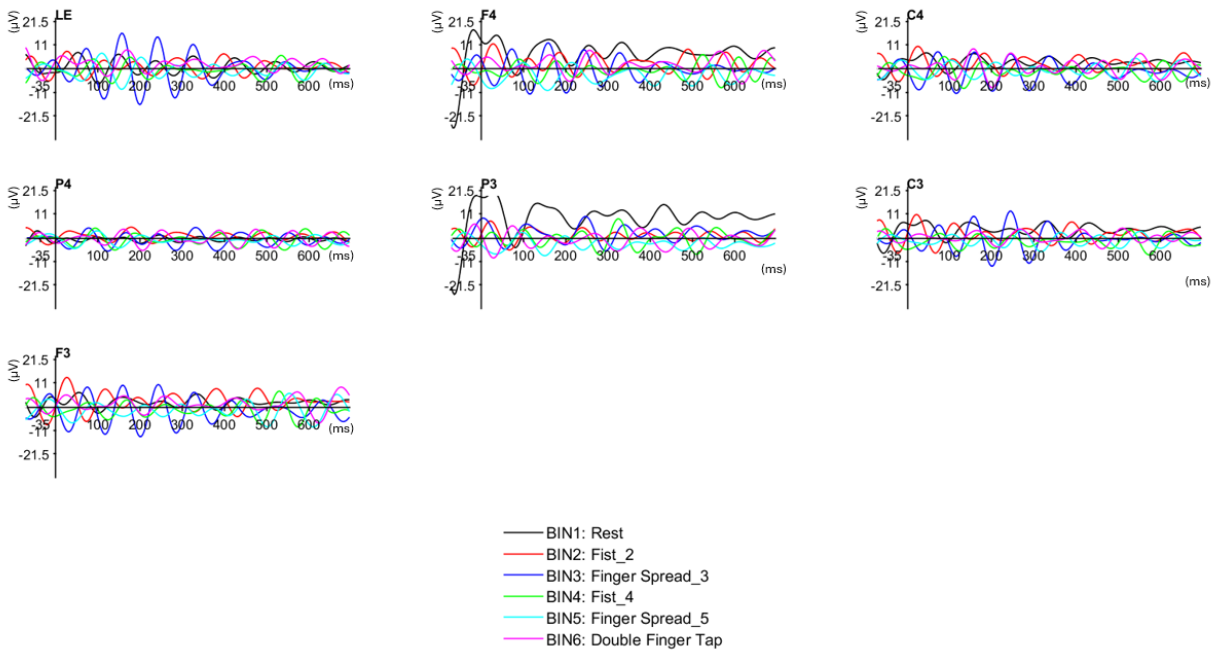


Figure 5.14: ERPSet for one subject's session for Game Flow 1

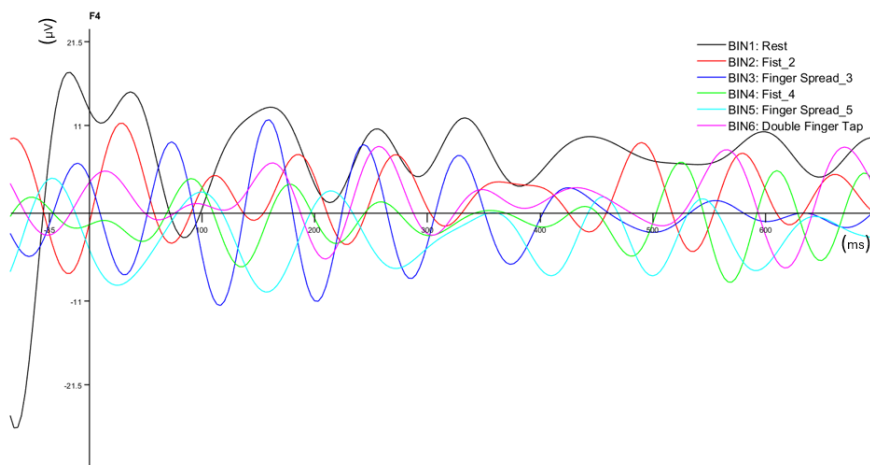


Figure 5.15: ERP for channel F4 for one subject's session for Game Flow 1

The figure 5.16 displays the ERP scalp maps for all bins present in Game Flow 1. The ERP scalp maps display brain activity at different scalp locations in response to specific stimuli or events, labeled as Bin1 through Bin6. Each map is color-coded, with dark blue indicating lower activity and yellow indicating higher activity. The spatial distribution of activity varies across the scalp, with certain regions showing higher activity, particularly in the central and frontal areas, which are often associated with cognitive and motor functions. The variability in activity patterns across different bins suggests that different stimuli or conditions elicit distinct neural responses. In the image, “0 ms” marks the start of time, right when hand movements begin. Each map shows the brain’s response at this moment for different hand movements such as Rest, Finger Spread, Fist, and Double Finger Tap.

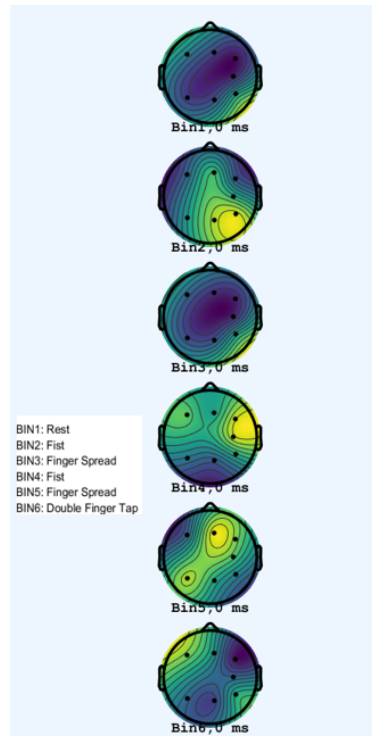


Figure 5.16: ERP Scalp Maps for one subject’s session for $\mathbb{G}_1$

We examined component ERPs mapped to corresponding Independent Components from the ICA process to evaluate hand gesture results. Figure 5.17 displays clear ERP compo-

nents, including peaks around 100 ms and 300 ms, likely representing the P100 and P300 components for sensory and motor processing. Topographic maps (IC1 to IC7) illustrate the spatial distribution of brain activity, with varying colors indicating different levels of electrical potential, such as increased activity in the frontal region for IC1 and in the occipital region for IC3.

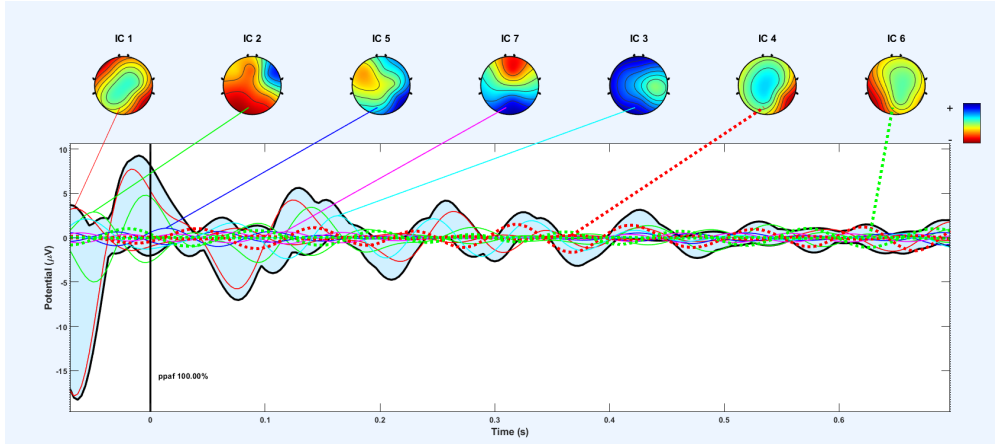


Figure 5.17: Component ERPs with component map for one subject's session for G_1

5.3.2 EMG Preprocessing Results

EMG preprocessing involved resampling the IMU gyroscopic and accelerometer data to match the frequency of the EMG data. Following this, a band-pass filter was applied to the EMG data, while a low-pass filter was used for the IMU data. The figure 5.18 displays the filtered EMG data. The chosen frequency range (20 Hz to 450 Hz) is common for EMG analysis. Frequencies below 20 Hz often contain baseline noise and movement artifacts, while frequencies above 450 Hz may include high-frequency noise or electrical interference. The values are typically in microvolts (μV) and represent the electrical activity of muscles. Positive values indicate muscle activation, while negative values indicate relaxation. The timing of these peaks or positive values varies in the figure, suggesting different muscle actions or movements.

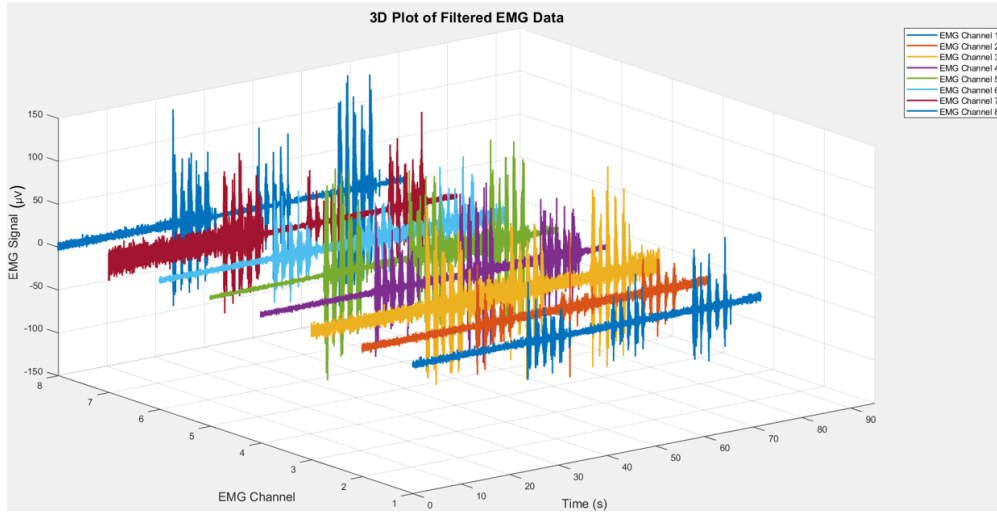


Figure 5.18: 3D Plots of each Channel of Filtered EMG Data for sample session for one volunteer in Game Flow 1. It is Frequency Filtered using a band pass filter of 20 – 450 Hz

After filtering the data, we segmented it and calculated features for both the EMG and IMU signals. The features extracted from the EMG and IMU data played a crucial role in classifying hand gestures by capturing essential characteristics of the signals. For EMG data, the mean absolute value (MAV) measured the average amplitude of the signal, which helped distinguish different muscle activations associated with various hand gestures. The root mean square (RMS) provided a measure of the signal's power, capturing the overall strength of muscle contractions, while the variance indicated the consistency of muscle activation, aiding in differentiating gestures with more or less variable signals.

For IMU data, the gyroscopic mean measured the average angular velocity around the x, y, and z axes, capturing the rotational movements involved in different gestures. The gyroscopic standard deviation measured the variability in angular velocity, indicating the consistency of these rotations. Similarly, the accelerometer mean measured the average acceleration along the x, y, and z axes, helping identify different translational movements. The standard deviation of the accelerometer data measured the variability in acceleration, providing insights into the consistency of translational movements.

We observed that combining these features allowed the classification model to understand the distinct patterns associated with each hand gesture. The EMG features provided insights into muscle activity, while the IMU features captured the dynamics of hand movements. Together, these features created a comprehensive representation of each gesture, enabling more accurate classification by the machine learning models.

5.3.3 Anomalous Artifact Findings

During preprocessing, we conducted a thorough evaluation of data quality. Initially, we identified and removed unwanted channels that were deemed unnecessary for our analysis. We then used Independent Component Analysis (ICA) to identify and discard components with significant artifacts. Specifically, components showing substantial contamination from heart signals, line noise, or eye blinks were removed. This step was crucial because heart signals and other artifacts can distort EEG data, leading to inaccurate results.

Our analysis revealed that, overall, the dataset was relatively clean. The primary artifacts we encountered were those related to heart signals, which were successfully identified and rejected.

Following the ICA process, we performed bin-based epochs, where the data was segmented into epochs around specific events or markers. During this phase, we also addressed any remaining artifacts. Typically, we discarded the signals collected before the first marker of rest, as these often exhibited high levels of noise.

The result of these preprocessing steps ensured that the data was as clean and accurate as possible, enhancing the reliability of subsequent analyses and interpretations.

5.4 Feature Engineering

After preprocessing the EEG and EMG data, we proceed with feature engineering to extract meaningful attributes from the dataset.

EEG Features In the extraction of EEG-ERP features, a range of statistical features is computed for each EEG channel (LE, F4, C4, P4, P3, C3, F3). These features include the mean (`col_mean`), standard deviation (`col_std`), minimum value (`col_min`), maximum value (`col_max`), range (`col_range`, calculated as the difference between maximum and minimum), median (`col_median`), skewness (`col_skew`), and kurtosis (`col_kurtosis`). Additionally, features related to the P300 component are extracted, including the peak value of the P300 (`col_P300_peak`) and the time at which this peak occurs (`col_P300_peak_time`) between 250 and 350 ms. Each of these features is calculated for every EEG channel and is appended to the data set along with the target label derived from the file name. Figure 5.22 displays the features used for the ERPs generated from the EEG data during one of the sessions in $\mathbb{G}_1$.

EMG Features Figure 5.28 shows the features of the EMG data collected during one of the $\mathbb{G}_1$ sessions. For the EMG data, we compute the Mean Absolute Value (MAV), Root Mean Square (RMS), Variance, Zero Crossing Rate (ZC), and Waveform Length (WL), utilizing all 8 emg channels. For the IMU data, we calculate the Mean and Standard Deviation, incorporating all 3 axes (x,y,z) of the IMU data.

5.5 Classification Results

Five Machine Learning (ML) models were chosen to evaluate the performance of the preprocessing techniques. The chosen models were Support Vector Machine (SVM), Convolutional Neural Network (CNN), Feedforward Neural Network (FNN), Recurrent Neural Network (RNN), and Long Short-Term Memory (LSTM). These models represent a significant range of machine learning models, each with strengths and weaknesses. Each task saw some variance in the effectiveness of the models and their performance in each category of metrics. With a small EEG and EMG dataset, I used data augmentation to create more samples. This helped improve the model's performance and reliability.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	LE_mean	LE_std	LE_min	LE_max	LE_range	LE_mediani	LE_skew	LE_kurtosis	F4_mean	F4_std	F4_min	F4_max	F4_range	F4_mediani	F4_skew	F4_kurtosis	C4_mean	C4_std	C4_min	C4_max	C4_range	C4_mediani	C4_sk
2	0.165947	4.760926	-8.24715	8.777823	17.02498	0.063604	0.00432	-1.35831	0.303214	4.32467	-8.66924	9.630615	18.29985	0.218531	0.064613	-0.62478	0.231785	3.46784	-7.056	7.48995	14.54595	0.004429	0.0047
3	0.02723	6.748778	-14.6045	14.32557	28.93004	-0.2819	0.005176	0.78476	-0.07887	5.72529	-12.6748	12.37486	25.04975	0.014877	0.022244	-0.41703	-0.09609	4.284193	-9.79177	9.483075	19.27484	-0.04701	0.0274
4	0.04646	5.502446	-12.069	12.64384	24.71285	-0.15072	-0.0188	-0.29304	0.170187	4.63098	-11.0534	15.90615	26.9596	0.0321	0.59296	1.701233	-0.01812	2.969504	-12.7483	9.769719	22.51801	-0.16397	-0.015
5	-0.17554	6.46109	-12.9853	13.01867	26.00401	-0.08857	0.068256	-0.72982	-0.06081	6.038576	-15.5196	14.4442	29.9638	-0.27532	0.041356	0.02235	-0.02518	3.860293	-8.34891	8.352512	16.70142	-0.00567	0.0105
6	0.117632	3.410146	-9.93933	8.973842	18.91317	-0.00536	0.02848	1.179423	0.115221	3.65423	-7.99923	7.061711	15.06094	-0.20325	-0.07576	-0.60421	0.106322	2.357349	-4.96933	4.931839	9.901173	-0.19171	0.001
7	-0.01997	1.974375	-3.43488	3.726582	7.161459	0.005827	0.017314	-1.16159	-0.04402	2.831526	-5.99206	5.00671	10.99877	-0.06837	-0.02459	-1.22393	0.050841	2.451645	-4.68067	4.673007	9.353677	0.089836	-0.056
8	0.11484	2.443273	-6.00477	6.837253	12.84202	0.323115	-0.03994	0.396818	-0.05131	2.392948	-5.62974	5.3019	10.93164	0.170105	-0.00971	-0.23528	-0.06598	1.564819	-3.37406	3.309412	6.683475	0.045111	0.0715
9	0.061683	3.030649	-5.27776	5.64202	11.16978	-0.09617	-0.03694	-1.13851	0.006461	2.699461	-5.55718	4.767825	10.32501	-0.02556	-0.05159	-1.00542	-0.00748	1.787958	-3.40322	3.576221	6.979436	0.009338	-0.015
10	0.126021	2.346209	-4.80089	5.952255	10.75314	0.066996	0.221597	-0.46279	-0.13698	3.330234	-7.67544	6.821368	14.49681	-0.64549	0.074881	-0.58801	-0.07749	2.474691	-6.72197	5.599665	12.32164	-0.17347	-0.061
11	-0.0336	2.33159	-5.47452	6.070765	11.54529	-0.11407	0.015837	0.105089	0.061462	2.805441	-6.35156	5.898605	12.25017	0.009623	-0.02044	-0.82232	-0.02573	1.583441	-3.32355	3.49185	6.815404	-0.08628	0.0298
12	0.031848	2.325608	-5.6648	5.430664	11.09546	0.069373	-0.02115	-0.48218	0.073951	2.377693	-5.74037	5.446853	11.18723	0.069657	0.020084	-0.2242	0.061623	1.822815	-3.37581	3.400991	6.7768	0.069217	-0.024
13	0.014336	1.081388	-4.27343	4.680784	8.954218	0.763134	0.760348	0.001076	0.6276	0.435333	0.435005	4.629149	4.662333	0.338881	0.081601	0.136241	0.136241	0.043064	0.174376	1.081334	4.130574	7.130608	0.0374

Figure 5.19: EEG-ERP Features - Part 1

	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS	AT
1	C4_kurtosis	P4_mean	P4_std	P4_min	P4_max	P4_range	P4_mediani	P4_skew	P4_kurtosis	P3_mean	P3_std	P3_min	P3_max	P3_range	P3_mediani	P3_skew	P3_kurtosis	C3_mean	C3_std	C3_min	C3_max	C3_range	C3_med
2	-0.8453	0.131119	2.674644	-6.72843	6.630831	13.35926	0.006771	0.005868	0.01886	0.045094	1.938932	-3.75876	4.014405	7.773161	0.075226	-0.07149	-0.95864	0.268055	3.681211	-7.44122	7.552546	14.99376	0.1957
3	-0.44825	-0.13785	2.271242	-5.64761	6.075844	11.72346	-0.22247	0.066089	0.194639	-0.02761	3.356364	-7.03652	7.153265	14.18978	-0.16107	0.038569	-0.55554	0.111821	6.300971	-12.3157	12.84004	25.15574	0.1647
4	3.606722	-0.10393	1.983527	-4.76579	3.861799	8.62759	0.008645	-0.06291	-0.47989	0.045589	2.898805	-6.47451	6.6234	13.09791	0.054035	0.055468	-0.73374	0.111087	3.843493	-8.12656	8.539796	16.66636	-0.111
5	-0.6771	0.005658	1.963339	-4.87271	4.666875	9.539584	0.113093	0.016211	-0.0012	-0.14682	1.977392	-4.18122	4.304088	8.521303	-0.10052	0.022432	-0.61335	-0.1532	4.19832	-9.4557	8.966659	18.15236	-0.204
6	-0.5217	0.008889	2.052749	-3.53368	4.542157	8.075834	0.171548	0.070363	-1.08995	-0.04513	1.294347	-2.93324	2.973253	5.90649	-0.13789	0.093929	-0.68905	0.11007	2.271666	-4.99546	4.34319	9.338651	0.2997
7	-1.07073	0.021941	1.207709	-2.83007	2.576995	5.407066	0.083826	-0.03655	-0.73422	0.001999	0.797437	-1.52598	1.45998	2.985959	0.099181	-0.07059	-1.17608	0.037708	2.412254	-4.61092	4.319947	8.930867	-0.081
8	-0.8693	-0.06437	2.200591	-4.09479	4.396663	8.491451	-0.08908	0.064728	-1.12755	-0.00681	2.077551	-3.82393	3.740095	7.564028	0.1039	-0.01596	-1.17336	-0.05079	2.507385	-6.06876	6.863121	12.93188	-0.107
9	-1.11477	-0.00108	2.251813	-5.76998	5.015303	10.78529	-0.10281	-0.00608	-0.17094	-0.0207	1.106372	-2.0503	2.381449	4.431746	0.006506	0.006974	-0.88805	0.039428	2.449096	-5.02125	5.407672	10.42892	0.1821
10	-0.18703	0.051006	3.019003	-6.2226	5.957023	12.17963	0.093028	-0.04617	-0.92242	-0.06766	1.97514	-4.97984	3.625423	8.60526	0.107944	-0.2971	-0.64411	-0.01211	3.000131	-6.47211	6.527787	12.9999	0.0857
11	-0.6804	0.011061	2.060405	-4.60052	5.128258	9.728777	0.000887	-0.04129	-0.23031	0.017972	1.935095	-4.10393	3.992633	8.096558	0.033349	-0.04107	-0.84824	0.107084	1.790016	-5.01425	5.962976	10.97722	0.0107
12	-0.10192	0.021397	1.424454	-2.73493	2.801122	5.536052	0.002791	0.023472	-0.79728	0.047748	1.046701	-2.28665	2.320096	4.607554	0.015932	-0.0581	-0.0871	0.018873	1.87297	-3.78982	3.959518	7.385042	0.1147
13	-0.87436	0.09163	1.386886	-2.01945	6.854273	8.873721	-0.04962	1.83202	6.348126	0.094826	1.380268	-2.73236	5.457955	8.190315	0.071828	0.572732	1.439418	0.080535	1.667706	-3.64516	5.242007	8.887167	0.0511
14	-1.19232	0.018236	1.763941	-3.87892	3.82055	7.698466	0.082272	-0.03736	-0.56811	0.114519	2.328325	-4.33614	4.572028	8.908165	0.122522	-0.05045	-0.95452	0.21088	0.056977	-8.38669	7.473553	15.86024	0.4027
15	-0.46237	0.011748	1.779827	-3.6341	3.460808	7.094907	0.067979	0.038459	-0.92917	0.00469	1.727565	-4.45397	4.251875	8.705842	-0.11864	0.012025	0.216466	-0.02876	3.70666	-9.40915	10.45951	19.86866	-0.077
16	0.034743	-0.00084	1.871317	-4.73952	4.07319	8.812706	-0.00401	-0.03212	-0.15935	-0.06118	3.095348	-5.62184	5.661624	11.28346	-0.18441	0.035342	-1.16264	-0.09399	4.077117	-9.45098	10.56446	20.01544	-0.34

Figure 5.20: EEG-ERP Features - Part 2

	AW	AX	AY	AZ	BA	BB	BC	BD	BE	BF	BG	BH	BI	BJ	BK	BL	BM	BN	BO	BP	BQ	BR	BS	B
1	_mean	F3_std	F3_min	F3_max	F3_range	F3_mediani	F3_skew	F3_kurtosis	P300_pei	P300_peak	P300_peak_j	P300_peak_j	P300_peak_j	P300_peak_j	P300_peak_j	P300_peak_j	P300_peak_j	P300_peak_j	P300_peak_j	P300_peak_j	P300_peak_j	P300_peak_j	P300_peak_j	target
2	0.2687	3.801894	-6.59296	8.063853	14.65681	0.488412	0.077799	-0.7505	7.680914	340	4.489478	250	6.076125	340	2.994782	340	2.016235	250	0.760641	266.6667	3.789702	350	Double Finger T	
3	169581	6.929119	-14.0269	14.20574	28.23262	0.396133	-0.03358	-0.81793	14.32557	313.3333	12.37496	316.6667	6.483075	316.6667	1.685918	313.3333	6.464338	313.3333	11.52426	316.6667	14.20574	316.6667	Finger Spread_3	
4	149272	5.261588	-10.35	13.51579	23.86578	0.080772	0.155627	-0.60299	9.909261	253.3333	4.467565	343.3333	1.740376	333.3333	3.561472	313.3333	3.390116	250	5.803498	350	7.445504	350	Finger Spread_5	
5	131246	5.225415	-11.2979	11.15284	22.45078	-0.15695	0.085538	-0.38259	13.01867	306.6667	12.74016	300	6.084826	293.3333	3.944141	250	1.056983	250	7.695606	300	11.15284	303.3333	Fist_2	
6	088069	3.625883	-9.36705	8.315784	17.68284	-0.12774	-0.06296	-0.0531	1.446116	350	5.39519	310	3.90392	300	2.970865	286.6667	0.958928	313.3333	3.988289	310	4.925668	310	Fist_4	
7	0.01884	2.658637	-4.49232	4.409875	8.902192	0.078801	-0.01696	-1.39549	2.488497	320	4.822803	323.3333	4.584398	320	2.194299	316.6667	0.666906	313.3333	2.32681	326.6667	3.912163	323.3333	Rest	
8	053287	2.801124	-5.98555	5.356863	11.34422	-0.10473	-0.04073	-0.10103	2.770588	273.3333	1.61232	290	0.979002	250	3.001672	336.6667	3.306347	270	4.783636	316.6667	3.9327	293.3333	Double Finger T	
9	002513	2.490797	-5.2617	5.706108	10.9678	-0.23634	0.017693	-0.61032	4.097858	303.3333	4.425137	300	2.673204	290	4.50476	290	2.381449	270	4.487962	276.6667	5.706108	283.3333	Finger Spread_3	
10	133057	4.125365	-8.21547	8.098765	16.31423	0.242952	-0.05618	-0.96688	4.545667	283.3333	4.33877	293.3333	3.962148	300	4.62461	303.3333	2.932108	300	3.840408	273.3333	4.330827	343.3333	Finger Spread_5	
11	072416	2.206477	-5.10945	4.944714	10.05416	0.097562	-0.12689	-0.4062	1.430852	296.6667	2.819548	346.6667	3.49185	343.3333	2.5845	336.6667	3.992633	350	2.218041	250	1.065346	276.6667	Fist_2	
12	088743	2.37701	-6.01682	6.886522	12.90334	-0.09501	0.224539	0.648888	1.791524	290	2.271004	300	2.285944	316.6667	1.224324	303.3333	1.284171	260	1.84466	250	1.053735	250	Fist_4	
13	031087	2.007409	-5.31174	4.741777	10.05313	0.177003	-0.03627	-0.61882	2.958408	296.6667	3.333333	314.0791	2.733333	345.867	1.1381	296.6667	1.08804	250	1.747169	266.6667	2.050328	266.6667	Rest	

Figure 5.23: EMG Features
- Part 1

Figure 5.24: EMG Features
- Part 2

Figure 5.25: EMG Features
- Part 3

Figure 5.26: EMG Features
- Part 4

Figure 5.27: IMU Features of Mean and Standard Deviation with Label

Figure 5.28: Overall EMG Features

5.5.1 Binary Classification

We will discuss the binary classification of rest versus motor tasks and within different motor tasks. Table 5.1 presents the binary classification results for various motor tasks, highlighting the best-performing machine learning model among the five evaluated—SVM, CNN, FNN, RNN, and LSTM—based on statistical metrics. For these classification calculations, data and features from EEG and EMG features were used for both Game Flow 1 ($\mathbb{G}_1$) and Game Flow 2 ($\mathbb{G}_2$). The overall results of binary classifications for all motor tasks is in the section of Appendix B. We observe that CNN and FNN performed the best for most of the binary classification tasks. SVM also did well but wasn't as strong as CNN and FNN. RNN and LSTM, on the other hand, didn't perform as well, likely due to differences in their architecture and how well they handle the data features. Additionally, the small amount of data limited our ability to train the models effectively, so we had to use data augmentation. The limited amount of data raises the risk of overfitting. This means the models might do well on the training data but struggle with new, unseen data. Although we used 5-fold cross-validation to test the models, there's still a chance that overfitting could occur.

Table 5.1: ML Models Metrics for Binary Classification

Motor Tasks (Down)/Metrics (Right)	Best Accuracy	Best Precision	Best Recall	Best F1 Score
Rest vs Fist	CNN (0.9766)	CNN (0.9756)	CNN (0.9766)	CNN (0.9756)
Rest vs Finger Spread	FNN (0.9445)	FNN (0.9502)	FNN (0.9445)	FNN (0.9443)
Rest vs Hand Wave In	CNN (0.9467)	FNN (0.9589)	CNN (0.9467)	SVM (0.9451)
Rest vs Hand Wave Out	CNN (0.9558)	CNN (0.9550)	CNN (0.9558)	CNN (0.9539)
Rest vs Double Finger Tap	CNN (0.9565)	CNN (0.9649)	CNN (0.9565)	CNN (0.9581)
Fist vs Finger Spread	CNN (0.9246)	FNN (0.9253)	FNN (0.9062)	CNN (0.9187)
Fist vs Hand Wave In	CNN (0.9565)	CNN (0.9649)	CNN (0.9473)	CNN (0.9581)
Fist vs Hand Wave Out	FNN (0.9470)	FNN (0.9577)	FNN (0.9432)	FNN (0.9481)
Fist vs Double Finger Tap	FNN (0.9495)	FNN (0.9592)	FNN (0.9433)	FNN (0.9493)
Finger Spread vs Hand Wave In	FNN (0.9169)	FNN (0.9253)	FNN (0.9188)	FNN (0.9147)
Finger Spread vs Hand Wave Out	CNN (0.9498)	CNN (0.9472)	CNN (0.9501)	CNN (0.9480)
Finger Spread vs Double Finger Tap	FNN (0.9495)	FNN (0.9578)	FNN (0.9433)	FNN (0.9496)
Hand Wave In vs Hand Wave Out	CNN (0.9867)	CNN (0.9886)	CNN (0.9867)	CNN (0.9868)
Hand Wave In vs Double Finger Tap	CNN (0.9478)	FNN (0.9545)	CNN (0.9478)	CNN (0.9493)
Hand Wave Out vs Double Finger Tap	FNN (0.9561)	FNN (0.9642)	FNN (0.9561)	FNN (0.9569)

In some cases, the accuracy and recall scores were the same, which means the classes were balanced or the model performed equally well (or poorly) on both measures. This balance probably happened because there were an equal number of samples for each motor task, as each task was performed for about the same amount of time.

5.5.2 Multi-Class Classification

In multi-class classification, we categorize the EEG and EMG data into multiple target classes rather than just two. In our experimental setup, we classify the data into six distinct hand movements: Rest, Fist, Finger Spread, Hand Wave In, Double Finger Tap, and Hand Wave Out. We will now analyze the results using both combined EEG and EMG features, as well as analyzing them separately.

When we evaluated ML models using only EEG features or only EMG features, the results were suboptimal, as evidenced by tables 5.2 and 5.3 .

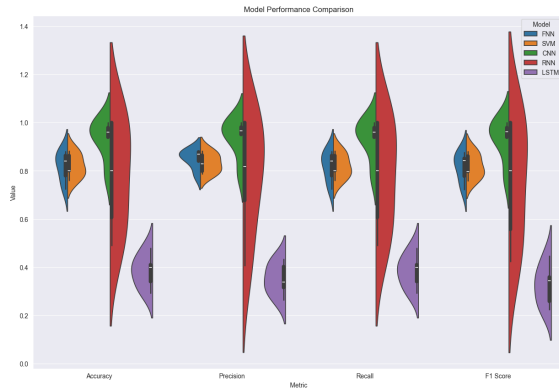
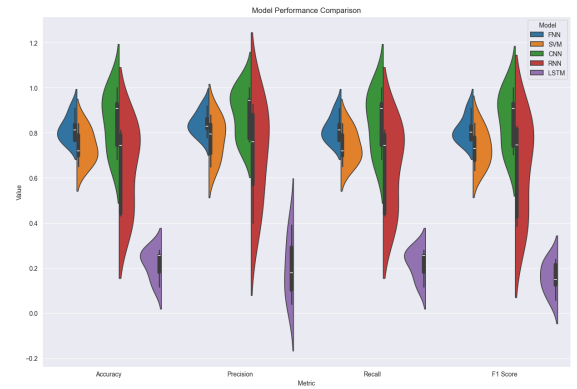
Using only EEG features: Table 5.2 shows the best performing ML model for game flow 1 ($\mathbb{G}_1$) and game flow 2 ($\mathbb{G}_2$) using only EEG features.

Figures 5.29 and 5.30 displays the violin plots for the classification metrics and the ML models used in this study for ($\mathbb{G}_1$) and ($\mathbb{G}_2$) respectively using EEG features only. The violin plots compares the performance of five models (RNN, SVM, CNN, FNN, and LSTM) across four metrics: Accuracy, Precision, Recall, and F1 Score. CNN performs best with high medians and narrow interquartile ranges (IQRs), indicating consistent performance. SVM follows closely, while FNN and RNN show more variability with wider IQRs. SVM performs better than RNN and LSTM but not as well as CNN and FNN. These insights help in selecting the most suitable model based on the specific metric that is most important for the application.

In Game Flow 1 ($\mathbb{G}_1$), the Convolutional Neural Network (CNN) performs the best, achieving the highest accuracy (0.8935), precision (0.9010), recall (0.8935) and F1 score (0.8866). The Support Vector Machine (SVM) and the Feedforward Neural Network (FNN) also show strong results, but fall short of CNN's performance. The Recurrent Neural Network

Table 5.2: ML Models Metrics for $\mathbb{G}_1$ and $\mathbb{G}_2$ using EEG features only

Metrics (Down) / ML Models (Right)	SVM	CNN	FNN	RNN	LSTM
For Game Flow 1 ($\mathbb{G}_1$)					
Accuracy	0.7380	0.8935	0.8176	0.6518	0.3934
Precision	0.7519	0.9010	0.8499	0.5963	0.3214
Recall	0.7380	0.8935	0.8176	0.6518	0.3934
F1 Score	0.7367	0.8866	0.8148	0.6122	0.2996
For Game Flow 2 ($\mathbb{G}_2$)					
Accuracy	0.7403	0.8526	0.8190	0.6445	0.2178
Precision	0.7741	0.8852	0.8424	0.7073	0.2005
Recall	0.7403	0.8526	0.8190	0.6445	0.2178
F1 Score	0.7332	0.8565	0.8158	0.6408	0.1570

Figure 5.29: Game Flow 1 ($\mathbb{G}_1$) ML Model Performance Comparison Violin Plot using EEG features onlyFigure 5.30: Game Flow 2 ($\mathbb{G}_2$) ML Model Performance Comparison Violin Plot using EEG features only

(RNN) and Long-Short-Term Memory (LSTM) models perform poorly, with notably lower metrics.

For Game Flow 2 ($\mathbb{G}_2$), CNN remains the top performer with an accuracy of 0.8526, precision of 0.8852, recall of 0.8526 and an F1 score of 0.8565. The SVM and FNN models continue to perform well but slightly less effectively than in ($\mathbb{G}_1$). This might be because "Fist" and "Finger Spread" were repeated twice in Game Flow 1, but only done once in Game Flow 2. The RNN and LSTM models again show limited effectiveness.

Overall, the CNN is the most effective model across both game flows, while the SVM and FNN are also strong contenders. The RNN and LSTM consistently underperform, indicating they may not be suitable for this task.

Using only EMG features: Table 5.3 shows the best performing ML model for game flow 1 ($\mathbb{G}_1$) and game flow 2 ($\mathbb{G}_2$) using EMG features only. Figures 5.31 and 5.32 displays

Table 5.3: ML Models Metrics for $\mathbb{G}_1$ and $\mathbb{G}_2$ using EMG features only

Metrics (Down) / ML Models (Right)	SVM	CNN	FNN	RNN	LSTM
For Game Flow 1 ($\mathbb{G}_1$)					
Accuracy	0.7159	0.8798	0.9103	0.8047	0.7540
Precision	0.7897	0.8810	0.9097	0.7927	0.7501
Recall	0.7159	0.8798	0.9103	0.8047	0.7540
F1 Score	0.5993	0.8721	0.9091	0.7846	0.7104
For Game Flow 2 ($\mathbb{G}_2$)					
Accuracy	0.8043	0.9305	0.9568	0.8195	0.7459
Precision	0.8022	0.9275	0.9571	0.8131	0.7442
Recall	0.8043	0.9305	0.9568	0.8195	0.7459
F1 Score	0.7774	0.9268	0.9563	0.7961	0.6822

the violin plots for the classification metrics and the ML models used in this study for ($\mathbb{G}_1$)

and $(\mathbb{G}_2)$ respectively using EMG features only. The violin plots compares the performance of five models (RNN, SVM, CNN, FNN, and LSTM) across four metrics: Accuracy, Precision, Recall, and F1 Score. FNN performs best with high medians and narrow interquartile ranges (IQRs), indicating consistent performance. CNN and RNN show more variability with wider IQRs. RNN performs better than SVM and LSTM but not as well as CNN and FNN.

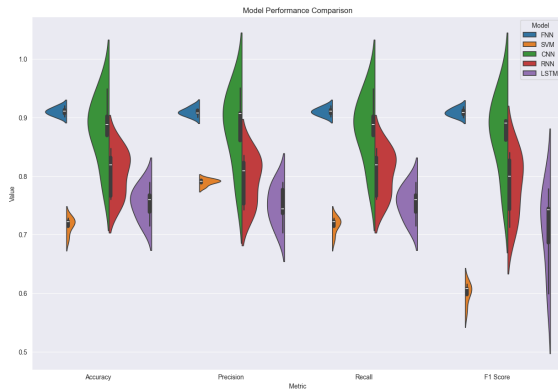


Figure 5.31: Game Flow 1 ($\mathbb{G}_1$) ML Model Performance Comparison Violin Plot using EMG features only

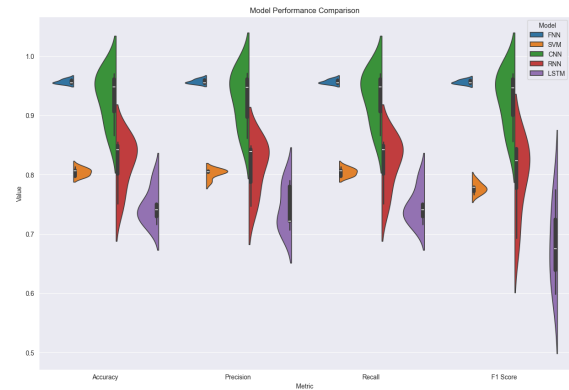


Figure 5.32: Game Flow 2 ($\mathbb{G}_2$) ML Model Performance Comparison Violin Plot using EMG features only

For Game Flow 1, the FNN model stands out with the highest accuracy (0.9103), precision (0.9097), recall (0.9103), and F1 score (0.9091). CNN also performs well, with statistical metrics values close to those of FNN. On the other hand, SVM shows the lowest performance across all metrics for Game Flow 1, with an accuracy of 0.7159 and an F1 score of 0.5993.

For Game Flow 2, FNN again demonstrates superior performance, achieving the highest accuracy (0.9568), precision (0.9571), recall (0.9568), and F1 score (0.9563). CNN follows closely, with strong performance across all metrics. In contrast, LSTM shows the lowest performance for Game Flow 2, with an accuracy of 0.7459 and an F1 score of 0.6822.

Overall, FNN consistently outperforms other models across all metrics for both Game Flow 1 and Game Flow 2, making it the most effective model when we are only using EMG features. CNN also performs well but slightly below FNN. SVM and LSTM show lower

performance compared to FNN and CNN, with SVM having the lowest scores for Game Flow 1 and LSTM for Game Flow 2. RNN performs moderately well but does not reach the high performance levels of FNN and CNN.

Combined Features Classification: Before applying the ML models, we combined the features of the EEG and EMG data of each session in $\mathbb{G}_1$ and $\mathbb{G}_2$ separately, which created a richer set of features for training. This enhanced feature set was then labeled and used to train the classification models. We found that incorporating the **combined features** significantly improved the ability of the models to classify motor tasks, providing more accurate results. Table 5.4 shows the best performing ML model for game flow 1 ($\mathbb{G}_1$) and game flow 2 ($\mathbb{G}_2$) with combined features of EEG and EMG.

Table 5.4: ML Models Metrics for $\mathbb{G}_1$ and $\mathbb{G}_2$ with combined features from EEG and EMG

Metrics (Down) / ML Models (Right)	SVM	CNN	FNN	RNN	LSTM
For Game Flow 1 ($\mathbb{G}_1$)					
Accuracy	0.9	0.9563	0.9	0.4688	0.2125
Precision	0.9238	0.9705	0.9229	0.5367	0.0457
Recall	0.9	0.9563	0.9	0.4688	0.2125
F1 Score	0.9005	0.9556	0.8981	0.4616	0.0751
For Game Flow 2 ($\mathbb{G}_2$)					
Accuracy	0.8139	0.9756	0.8480	0.4130	0.1573
Precision	0.8490	0.9792	0.8855	0.4415	0.0270
Recall	0.8139	0.9756	0.8480	0.4130	0.1573
F1 Score	0.8129	0.9737	0.8494	0.4015	0.0455

For Game Flow 1 ($\mathbb{G}_1$), the Convolutional Neural Network (CNN) is the top performer, achieving an accuracy of 95.63% and an F1 score of 95.56%. It effectively balances positive

case identification and false positives, outperforming other models such as SVM, FNN, and showing much better performance than RNN and LSTM.

In Game Flow 2 ($\mathbb{G}_2$), CNN again excels with an accuracy of 97.56% and an F1 score of 97.37%, making it the optimal choice. SVM and FNN also performed well, but lagged behind CNN, with SVM with 81.39% accuracy and 81.29% F1 score and FNN with 84.80% accuracy and 84.94% F1 score. SVM's precision and recall align well with the CNN, although it falls short in overall performance. The RNN showed moderate results (41.30% precision and 40.15% F1 score), while the LSTM significantly underperformed with an accuracy of 15.73% and an F1 score of 4.55%.

In summary, CNN was the best performing ML model per classification metric for each of the game flows. Figures 5.33 and 5.34 displays the violin plots for the classification metrics and the ML models used in this study for ($\mathbb{G}_1$) and ($\mathbb{G}_2$) respectively.

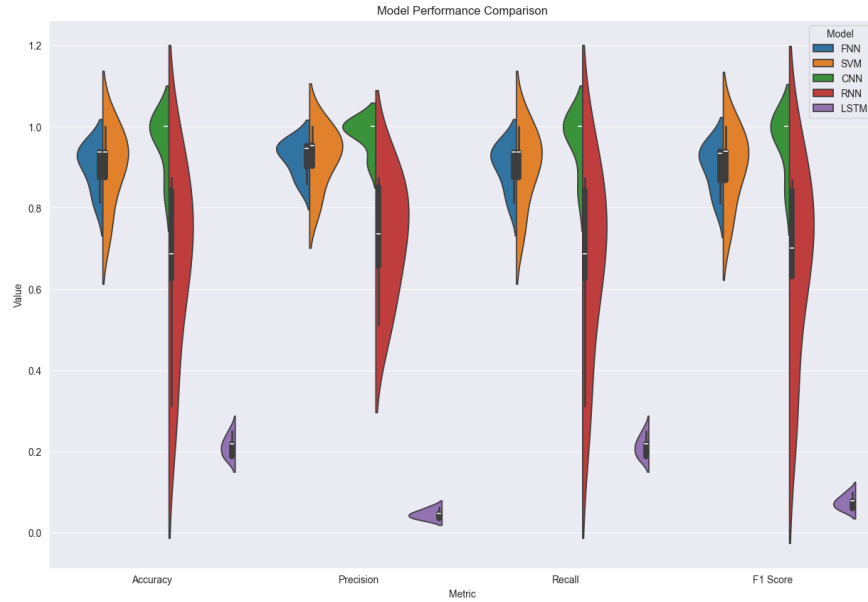


Figure 5.33: Game Flow 1 ($\mathbb{G}_1$) ML Model Performance Comparison Violin Plot

The violin plots compares the performance of five models (RNN, SVM, CNN, FNN, and LSTM) across four metrics: Accuracy, Precision, Recall, and F1 Score. CNN performs

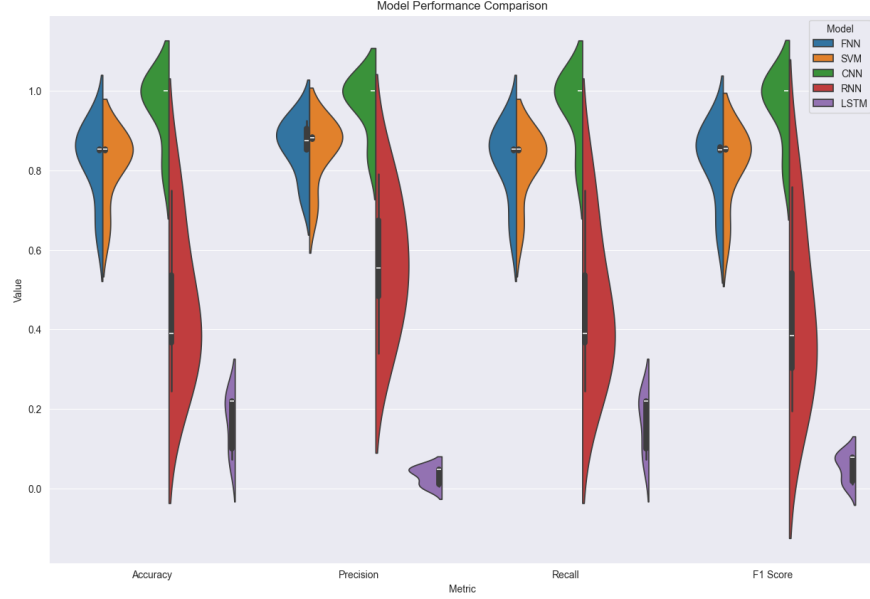


Figure 5.34: Game Flow 2 ($\mathbb{G}_2$) ML Model Performance Comparison Violin Plot

best with high medians and narrow interquartile ranges (IQRs), indicating consistent performance. FNN follows closely, while RNN and SVM show more variability with wider IQRs. SVM performs better than RNN and LSTM but not as well as CNN and FNN.

5.5.3 Statistical Analysis

Figure 5.35 shows the correlation matrix heat map between all the columns of the combined dataset of EEG and EMG features. The heat map shows that EEG and EMG features are generally not strongly related, but there are still some features that are connected. Here, **Values close to 1** indicate a strong positive correlation, **Values close to -1** indicate a strong negative correlation, and **Values around 0** indicate little to no linear correlation. The correlation coefficient ranges from -1 to +1, with the following meaning:

- **1**: Perfect positive correlation. As one variable increases, the other variable also increases proportionally.

- **-1**: Perfect negative correlation. As one variable increases, the other variable decreases proportionally.
- **0**: No correlation. There is no linear relationship between the variables.

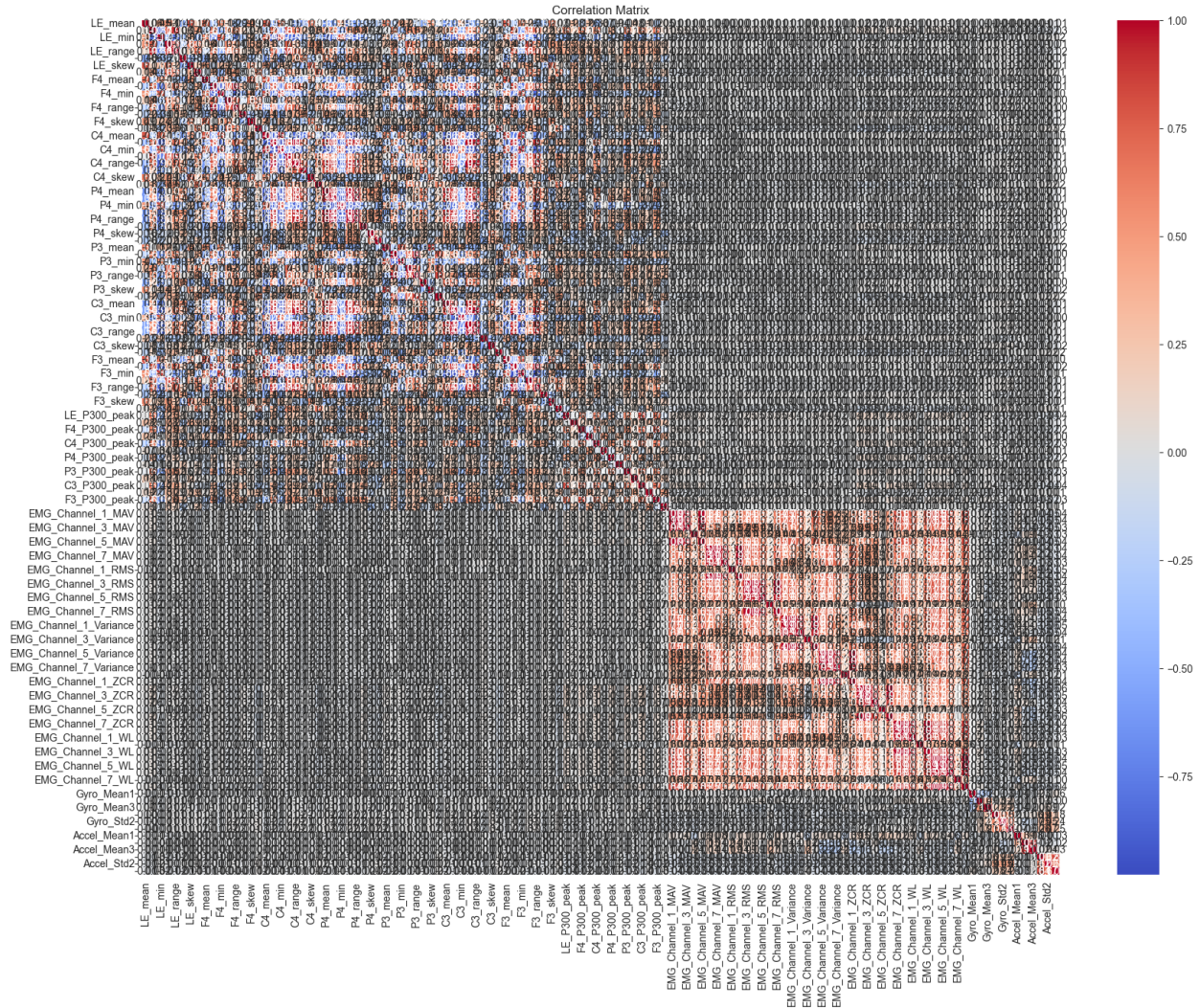


Figure 5.35: Correlation Matrix Heat Map

We can see that the correlation coefficient between the EMG channels of Zero Crossing Rate (ZCR) and P3_skew is around 0.6, indicating a moderate positive correlation. Additionally, there is a noticeable blue line for all EMG features related to F3_skew, suggesting

a consistent pattern. On the other hand, the RMS and Variance of the EMG channels show almost no correlation with P4_min. These observations highlight some of the key correlations revealed by the heat map.

ANOVA Analysis: The box plot figure 5.36 shows the anova analysis for the groups of features when only EEG features were used, and when only EMG features were used, and when they were used together. ANOVA F-value (1.3485): This value measures the ratio of variance between the group means to the variance within the groups. A higher F-value indicates a greater difference between group means. ANOVA p-value (0.2650): This value helps to determine the statistical significance of the observed differences. A p-value less than 0.05 typically indicates significant differences between groups. The box plot compares the out-

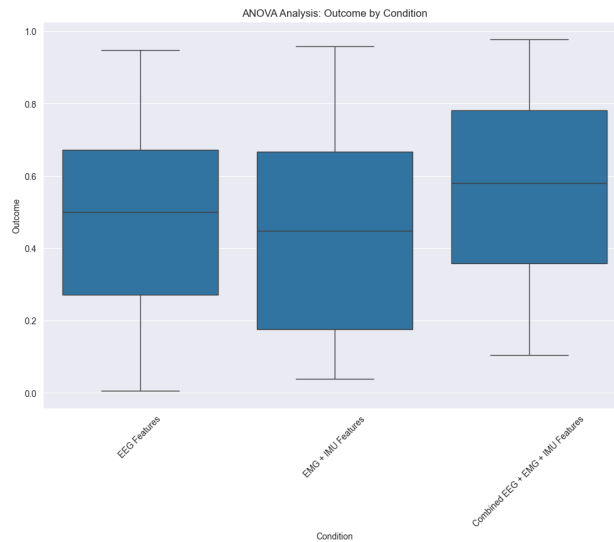


Figure 5.36: Anova Analysis per feature group

comes across three conditions: “EEG Features,” “EMG + IMU Features,” and “Combined EEG + EMG + IMU Features.” The conditions represent the use of only EEG features, only EMG features, or a combination of both. The y-axis represents the outcome, which is the prediction result given by the machine learning models, ranging from 0 to 1.

From the graph, we can observe the following:

- **EEG Features:** This condition shows a relatively wide interquartile range (IQR), indicating variability in the outcomes. The median result is around 0.6.
- **EMG + IMU Features:** This condition has a narrower IQR compared to EEG Features, suggesting more consistent outcomes. The median result is slightly higher, around 0.7.
- **Combined EEG + EMG + IMU Features:** This condition shows the highest median outcome, close to 0.8, with a relatively narrow IQR, indicating both high and consistent performance.

Overall, the combined features condition appears to give the best and most consistent outcomes, followed by EMG + IMU Features, and then EEG Features. This suggests that integrating multiple types of data (EEG, EMG, and IMU) can enhance the performance of our analysis.

5.6 Summary

Effect of Preprocessing Technique Configurations. The aim of this experiment was to explore how gamification can enhance rehabilitation for neurological conditions, specifically for patients with upper limb weakness, using EEG/EMG feedback. The results indicate that Independent Component Analysis (ICA) and Frequency Filtering (FF) significantly improved classification metrics, while Event-Related Potential (ERP) features had a more modest impact on reducing noise. Data segmentation in EMG was also a crucial step.

Combining FF, ICA, and ERP was found to be the most effective approach overall. Additionally, using filtered EMG data proved beneficial and incorporating IMU data enhanced the machine learning models' ability to detect hand movements, leading to increased accuracy.

These findings suggest that effective preprocessing can significantly boost classification accuracy, thereby enhancing the game's ability to provide valuable feedback for improving upper-limb function.

Effect of Machine Learning Model. Another goal of this experiment was to introduce statistical and ML techniques to analyze EEG and EMG data to study emerging patterns that can then be used to create adaptive simulation routines. Among the machine learning models evaluated, the convolutional neural network (CNN) stands out as the top performer. The Support Vector Machine (SVM) and the Feedforward Neural Network (FNN) also performed well, but the Recurrent Neural Network (RNN) and the Long Short-Term Memory (LSTM) models fell short in comparison.

Effect of 3D Simulation Results. We started with a basic 3D MR simulation to test the edge computing system as seen in the section of Appendix C. It engaged users effectively and ran smoothly on a Windows laptop, with potential for adaptation to Android and iOS, improving portability and accessibility. This simulation can be viewed as a **proof-of-concept for a future game** designed to be more engaging and immersive for patients. By demonstrating successful user engagement and smooth performance, it lays the groundwork for developing a more advanced and interactive experience.

In conclusion, this experiment highlights the importance of choosing appropriate preprocessing techniques and machine learning models for classifying motor tasks in EEG and EMG data. The findings of this study can be useful and give a proof of concept for developing efficient and accurate BCI systems with AR / MR games for the rehabilitation of upper limb extremity weakness.

Chapter 6

CONCLUSION

This section takes a closer look at the accomplishments and valuable research contributions of this work. While recognizing acknowledged limitations, it also explores promising areas for future research..

6.1 Conclusion

The overall objective of this research was to explore and address some of the challenges faced by the “NeuroRehab Ecosystem” as it evolves as a platform to study and help with the rehabilitation of neurological conditions, specifically when it comes to upper limb extremities and the potential gamification of the rehabilitation. We have done so through the innovative integration of augmented and mixed reality to the collection and analysis of EEG and EMG data, ultimately leading to the creation of a Mixed Reality NeuroGame. With this simulation, we aim to provide an immersive experience for users, predominantly incorporating Augmented Reality components to craft a dynamic and engaging environment. We carefully designed the *game flows* to include several actions that align with physiotherapy exercises essential for upper limb extremities rehabilitation.

The development of this simulation is one of the main contributions resulting from this research, as it provides a platform to develop and study more such games or simulations, e.g, targeting the rehabilitation of other neurological conditions and/or in other parts of the body.

Another contribution of this work has been to set up an outline of the experiment design, e.g., through the definition of our *game flows*, we set up an agreeable strategy and framework to collect and analyze EEG and/or EMG data in the way required to grow the NeuroRehab

Ecosystem.

Throughout our experiments and 3D simulation testing, EEG and EMG signals were systematically collected for subsequent in-depth analysis. These data served as a basis for building an ML-based recommendation system with the explicit goal of helping to accelerate patient recovery. The outcomes of our ML models (especially CNN) for classification used for the recommendation system are encouraging, scoring an approx. 95% of accuracy in detecting hand gestures across our set of participants for both the game flows. Our methodology for analyzing and using the data collected and the use of ML models and their evaluation in the context of our study, sets a precedent of how we can use various statistical and ML techniques that can be extended to future research and developments in this area.

Beyond quantitative achievements, qualitative feedback from the experimental cohort further substantiated the positive impact of the simulation. As we ran experiments on volunteering teammates and collaborators, they provided feedback on the game design, computational aspects such as performance, ML strategies and testing, and therapeutic effectiveness of the exercises. The comments were overall positive and provided plenty of guidance to improve this research and considerations for future work. We understand that while these subjects are biased, they also have technical knowledge and understand the goals of our research and were able to provide a good basis of positive testing. However, it is understood more testing is needed with different types of subjects that are unfamiliar with our research, and/or are non tech savvy.

Reflecting on the learning trajectory of this thesis journey, valuable insights were gained regarding the efficacy of specific hand movements in the rehabilitation process. Some of the best results we obtained correspond to hand gestures such as wave-in (Flexion), wave-out (Extension), making a fist, spreading fingers, and double(two) finger tap. This recurring theme found in our results highlighted the efficacy of a repetitive flow of exercises that incorporates these gestures and can facilitate accelerated patient recovery in the future.

Looking for ways to improve the study, an area to explore is adding custom *game paths* based on the exercises (actions) recommended by a rehabilitation therapist. Letting patients

design their own exercise routines could make rehab more personal, better suited to their preferences and needs, and ultimately more empowering.

Designing the Right Machine Learning Pipeline

Designing the appropriate ML and feedback system proved to be a complex and challenging task. It required careful consideration of various factors, including selecting the right ML algorithms, ensuring accurate data processing, and developing an effective feedback mechanism in Unity and MATLAB. Balancing these elements to create a system that effectively supports and enhances the user's experience was both demanding and complex. We conclude that combining EMG features with EEG signals significantly improved our classification, achieving an approx. 95% accuracy for both game flows. CNN emerged as the most robust model with the added EMG data, outperforming SVM and FNN, which were previously the best when using only EEG data. Thus, CNN is the optimal model for our experiments.

6.2 Future Work

In this section, we discuss several future work opportunities, some of which are prompted by ways to improve the present work, and others to overcome or manage some of the limitations we faced through this journey.

Update Game - Create Custom Game Flow. Introducing a feature for users to create custom flows within the game represents another compelling prospect. Enabling individuals to tailor their exercise routines not only improves personalization but also empowers users to engage with rehabilitation activities that align more closely with their specific needs and preferences.

Machine Learning Exploration. Based on our promising results, expanding on the use and testing more/different ML classification signals for distinct hand gestures. Investigating which signals prove more adept at identifying specific gestures could provide valuable insights

into the nuances of gesture recognition, potentially optimizing the overall effectiveness of the rehabilitation process. Additionally, collecting more and more varied data is imperative. We proposed two *game flows* $\mathbb{G}_1$ and $\mathbb{G}_2$, for which testing with more subjects will be important, but adding more *game flows* that either test different sequences of the same *tasks* in $\mathbb{G}_1$ and $\mathbb{G}_2$ or with different *tasks* altogether, will be paramount in the next steps of our proposed NeuroGame. For consistency, hyperparameter tuning was not performed in this experiment. Nevertheless, the results suggest that tailoring parameters for each motor task and machine learning model could significantly improve performance.

Updating Hardware Equipment. Working with Thalmic Labs’ MyoBands proved challenging, especially since the company no longer supports them. Obtaining the necessary software was difficult, and alternative methods, like finding a public repository, were time-consuming if the software wasn’t pre-installed. Additionally, the EEG DSI-7 Headset, being outdated, required considerable time for diagnosis and use, often breaking down and presenting connectivity issues with the DSI Streamer.

Update Game - real-time feedback based on ML models. Classifying multiple motor and muscle tasks simultaneously using ensemble classifiers presents a significant challenge but offers substantial benefits. This approach would address the issue of confounding factors where current models might misinterpret or treat different tasks similarly. By employing ensemble methods, the system could better differentiate between overlapping muscle activities, improving accuracy. Additionally, incorporating real-time feedback into the Unity game would greatly enhance the user’s experience. Real-time feedback allows users to receive immediate responses to their actions, making the rehabilitation process more interactive and engaging, and potentially leading to better outcomes.

Expand Experiments for Improved Testing. While our ML results are encouraging, they also motivate us to improve the way we test our existing and future ML models. Specif-

ically, our experiments are designed for *positive testing* of these models, i.e., we look to detect the existence of changes in brainwaves when an action occurs or lack of changes in the absence of actions (True Positives and True Negatives). However, it is important to provide means for *negative testing* as well, i.e., to detect False Positives and False Negatives.

Testing the Future Game in clinical environment. Another promising direction for future research is applying this approach to real patients in clinical environment with neurological disorders. This would involve testing and adapting the system in clinical settings to assess its effectiveness and usability for individuals with various neurological conditions. By doing so, the approach could be refined based on real-world patient feedback, potentially leading to more tailored and effective rehabilitation solutions.

Use Database and Cloud Services. Creating a central database for all devices in an edge computing system consolidates data from multiple sources into a unified repository. This centralization simplifies data management, improves analysis by integrating diverse datasets, and enhances security through unified control measures. Overall, it streamlines operations and facilitates better decision-making across the system.

These proposed avenues for future work determine the ongoing potential for innovation and refinement in the realm of NeuroGaming: Innovative Rehabilitation for Upper Limb Neurologic Conditions Using Mixed-Reality Simulation Games and EEG/EMG Biofeedback. Each suggestion opens doors to further exploration, offering the opportunity to enhance the user experience, advance machine learning capabilities, and ultimately contribute to the evolving landscape of therapeutic technologies.

BIBLIOGRAPHY

- [1] A. Marushima H. Watanabe A. Zaboronok S. Watanabe A. Matsumura K. Suzuki Y. Matsumaru E. Ishikawa A. Anastasiev, H. Kadone. Supervised myoelectrical hand gesture recognition in post-acute stroke patients with upper limb paresis on affected and non-affected sides. page 8733, 2013.
- [2] N. Hashimoto A. Boyali. Spectral collaborative representation based classification for hand gestures recognition on electromyography signals. pages 11–18, 2016.
- [3] M. H. Purnomo W. R. Islamiyah A. F. Setiawan, A. D. Wibawa. Monitoring stroke rehabilitation re-learning program using eeg parameter: A preliminary study for developing self-monitoring system for stroke rehabilitation during new normal. pages 620–624, 2020.
- [4] J. Taeymans C. Schuster-Amft A. Kruse, Z. Suica. Effect of brain-computer interface training based on non-invasive electroencephalography using motor imagery on functional recovery after stroke - a systematic review and meta-analysis. page 385, 2020.
- [5] M. C . Brady P. Langhorne G. E. Mead J. Mehrholz F. van Wijck A. Pollock, S. E. Farmer. Interventions for improving upper limb function after stroke. page CD010820, 2014.
- [6] M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover, New York, 9th printing edition, 1972.
- [7] Arnaud Delorme (2024). EEGLAB (<https://github.com/scn/eeqlab>).
- [8] J. Olivan B. Kemp. European data format ‘plus’ (edf+), an edf alike standard format for the exchange of physiological data. pages 1755–1761, 2003.

- [9] Basicmedical Key. Anterior Forearm.
- [10] Nima Bigdely-Shamlo, Jonathan Touryan, Alejandro Ojeda, Christian Kothe, Tim Mullen, and Kay Robbins. Automated eeg mega-analysis i: Spectral and amplitude characteristics across studies. *NeuroImage*, 207:116361, 2020.
- [11] Ali Boyali and Naohisa Hashimoto. Spectral collaborative representation based classification for hand gestures recognition on electromyography signals. *Biomedical Signal Processing and Control*, 24:11–18, 2016.
- [12] Ali Boyali, Naohisa Hashimoto, and Osamu Matsumoto. Hand posture and gesture recognition using myo armband and spectral collaborative representation based classification. In *2015 IEEE 4th Global Conference on Consumer Electronics (GCCE)*, pages 200–201. IEEE, 2015.
- [13] C. Browne. Evaluating the effectiveness of preprocessing methods on motor classification accuracy in eeg data. *A Thesis, University of Washington Bothell, CSSE*, 2023.
- [14] D. Woolley R. Gassert N. Wenderoth C. Lambelet, M. Lyu. The ewrist - a wearable wrist exoskeleton with semg-based force control for stroke rehabilitation. pages 726–733, 2017.
- [15] M. C. Demir C. Tepe. Detection and classification of muscle activation in emg data acquired by myo armband. pages 178–183, 2020.
- [16] Rona Campbell, M Evans, M Tucker, B Quilty, Paul Dieppe, and JL Donovan. Why don’t patients do their exercises? understanding non-compliance with physiotherapy in patients with osteoarthritis of the knee. *Journal of Epidemiology & Community Health*, 55(2):132–138, 2001.
- [17] M. A. Castro-Alamancos. The motor cortex: a network tuned to 7-14 hz. 2013.
- [18] NumPy Developers. *NumPy*, 2024. Version 1.24.0.

- [19] SciPy Developers. *SciPy*, 2024. Version 1.11.0.
- [20] P. Aricò F. Aloise F. Leotta F. de Vico Fallani J. del R. Millán M. Molinari D. Mattia F. Cincotti, F. Pichiorri. Eeg-based brain-computer interface to support post-stroke motor rehabilitation of the upper limb. pages 4112–4115, 2012.
- [21] Noor Farhana, Ahmad Firdaus, Mohd Darmawan, and Mohd Faizal. Evaluation of boruta algorithm in ddos detection. *Egyptian Informatics Journal*, 24, 11 2022.
- [22] GeeksforGeeks. Violin plot for data analysis, 2024.
- [23] J. M. Lim C. H. Hwang K. koo H. L. Phan, T. H. Le. Effectiveness of augmented reality in stroke rehabilitation: A meta-analysis. page 1848, 2022.
- [24] Katie J Hughes, Nancy Salmon, Rose Galvin, Blathin Casey, and Amanda M Clifford. Interventions to improve adherence to exercise therapy for falls prevention in community-dwelling older adults: systematic review and meta-analysis. *Age and Ageing*, 48(2):185–195, 10 2018.
- [25] John D. Hunter and Matplotlib contributors. *Matplotlib*, 2024. Version 3.7.1.
- [26] K. Felix Navarro J. Garcia. The mobile rehapp(tm): an ar-based mobile game for ankle sprain rehabilitation. pages 1–6, 2014.
- [27] A. L. Valdivieso Caraguay M. E. Benalcázar J. P. Váscenez, L. I. Barona López. Hand gesture recognition using emg-imu signals and deep q-networks. page 9613, 2022.
- [28] JetBrains. *PyCharm*, 2024. Version 2024.1.3.
- [29] J. Paavilainen J. Hamari K. Alha, E. Koskinen. Why do people play location-based augmented reality games: A study on pokémon go. pages 114–122, 2020.
- [30] Philipp Kanske. Exploring executive attention in emotion : Erp and fmri evidence. *MPI Series in Human Cognitive and Brain Sciences*, v.106 (2008).

- [31] J.Niu N. Duan X. Yu Q. Li S. Yeh L. Zheng L. Liu, J. Cui. Design of mirror therapy system base on multi-channel surface-electromyography signal pattern recognition and mobile augmented reality. page 2142, 2013.
- [32] Blaser Larry. *The Gale Encyclopedia of Science*. Gale, Farmington Hills, MI, 5 edition, 2014.
- [33] M. A. Auais M. Bamdad, H. Zarshenas. Application of bci systems in neurorehabilitation: a scoping review. pages 355–364, 2015.
- [34] J. Fu Y. Sun Q. Fang M. Lin, J. Huang. A vr-based motor imagery training system with emg-based real-time feedback for post-stroke rehabilitation. pages 1–10, 2023.
- [35] Media Title. homunculus. Image. Accessed June 15 , 2024.
- [36] Merck manuals. Symptoms of upper limb weakness neurologic disorders.
- [37] Mixamo. Characters and Animation.
- [38] mtrresources.tripod. Muscles.
- [39] Nurse Key. The Musculoskeletal System.
- [40] postureguides. Carpal tunnel exercises.
- [41] D. Youer Q. Xiaoyan, L. Douzhe. P300 feature extraction based on parametric model and fastica algorithm. pages 585–589, 2009.
- [42] J. M. Santos R. Ramele, A. J. Villar. Eeg waveform analysis of p300 erp with applications to brain computer interfaces. page 199, 2018.
- [43] D. Dawar M. Kaur J. Pandian R. Sahonta C. S. Kumar M. Mahadevappa S. Singh, B. Tiwari. Wavelet and region-specific eeg signal analysis for studying post-stroke rehabilitation. pages 6251–6254, 2021.

- [44] J. Wilbur S. T. Larson. Muscle weakness in adults: Evaluation and differential diagnosis. pages 95–108, 2020.
- [45] scikit-learn developers. *scikit-learn*, 2024. Version 1.3.0.
- [46] A. Siddalingaswami. Architecting a cybersecurity framework for an edge computing medical application for stroke patient rehabilitation. *A Capstone Project, University of Washington Bothell, CSSE*, 2023.
- [47] Emmy M Sluijs, Gerjo J Kok, and Jouke van der Zee. Correlates of Exercise Compliance in Physical Therapy. *Physical Therapy*, 73(11):771–782, 11 1993.
- [48] K. Ueno C. Shiroma R. Morita Y. Naito R. Ishii T. Inamoto, M. Ueda. Enhanced recognition of amputated wrist and hand movements by deep learning method using multimodal fusion of electromyography and electroencephalography. page 680, 2022.
- [49] K. Ueno C. Shiroma R. Morita Y. Naito R. Ishii T. Inamoto, M. Ueda. Motor-related mu/beta rhythm in older adults: A comprehensive review. page 751, 2023.
- [50] D. J. Gaebler-Spira J. P. A. Dewald T. Sukal-Moulton, K. J. Krosschell. Motor impairments related to brain injury timing in early hemiparesis. part ii: abnormal upper extremity joint torque synergies. pages 24–35, 2014.
- [51] TensorFlow Team. *TensorFlow*, 2024. Version 2.12.0.
- [52] The Pandas Development Team. *Pandas*, 2024. Version 2.0.0.
- [53] The McGill Physiology Virtual Lab. Biomedical signals acquisition, EEG, introduction, 2022.
- [54] Unity Technologies. Unity, 2023. Game development platform.
- [55] Michael Waskom. *Seaborn*, 2024. Version 0.12.2.
- [56] Wearable Sensing. DSI-Streamer.

- [57] Wearable Sensing. EEG Electrodes.
- [58] Piotr Wierzgała, Dariusz Zapala, Grzegorz M. Wojcik, and Jolanta Masiak. Most popular signal processing methods in motor-imagery bci: A review and meta-analysis. *Frontiers in Neuroinformatics*, 12, 2018.
- [59] Wikipedia. Neurogaming.
- [60] A. Wang Y. Wang. Augmented reality based upper limb rehabilitation system. pages 426–430, 2017.

Appendix A

PREPROCESSING TECHNIQUE CONFIGURATION AND PLUGINS - EEGLAB

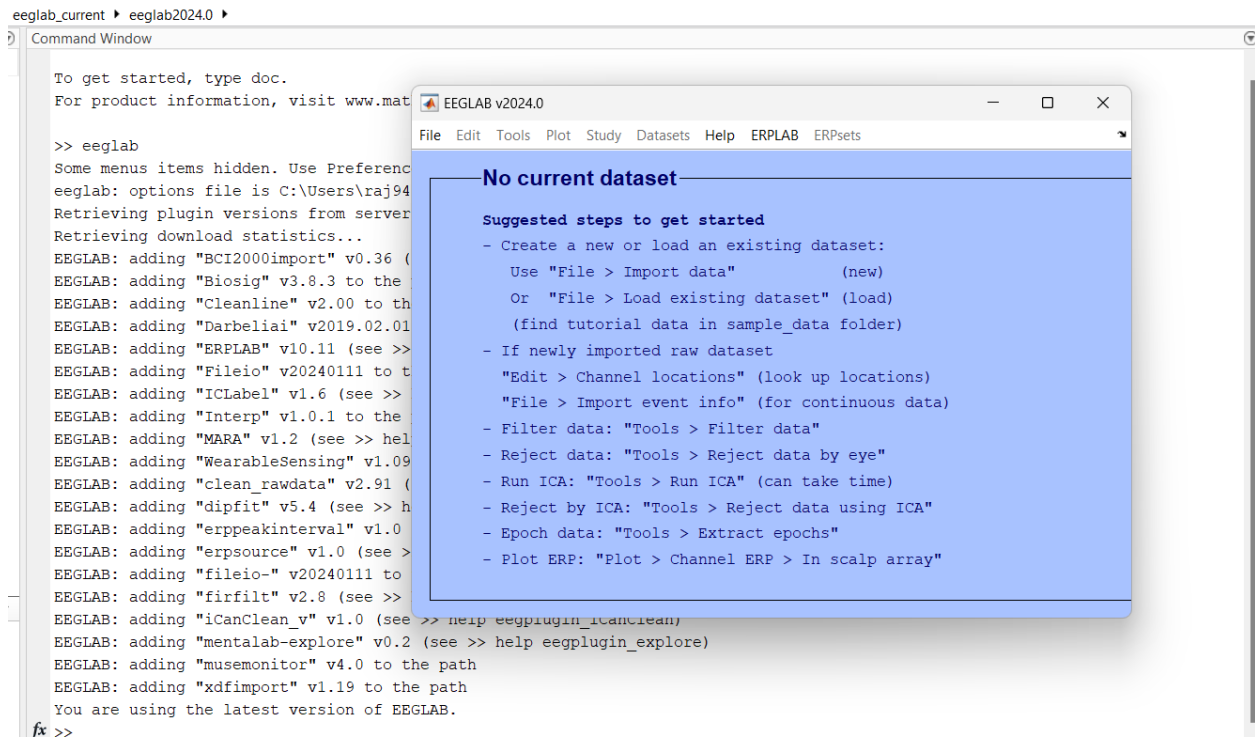


Figure A.1: EEGLAB opening window

EEGLAB Plugins installed while running the Preprocessing scripts: Biosig v3.8.3, Fileio v20240111, ERPLAB v11.03, clean_rawdata v2.91, Cleanline v2.00, ICLabel v1.6, firfilt v2.8, dipfit v5.4, xdfimport v1.19, MARA v1.2, BCI2000import v0.36, Darbeliai v2019.02.01.1, musemonitor v4.0, WearableSensing v1.09, erppeakinterval v1.0, erpsource v1.0, Interp v1.0.1

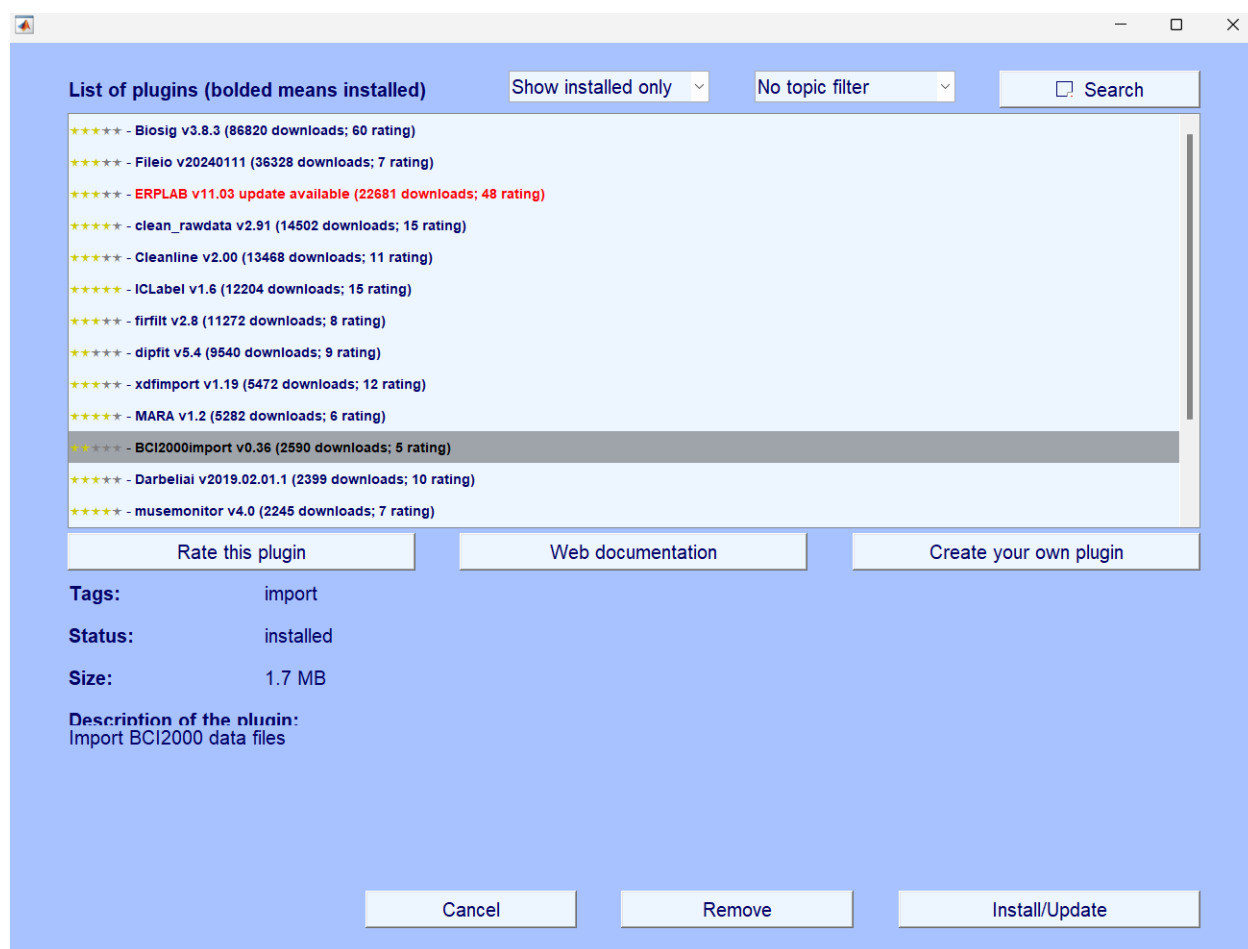


Figure A.2: EEGLAB Plugins Manager

Appendix B

BINARY CLASSIFICATION RESULTS

Table B.1: ML Models Metrics for Rest vs Fist

Metrics (Right) / ML Models (Right)	Accuracy	Precision	Recall	F1 Score
CNN	0.9766	0.9756	0.9766	0.9756
FNN	0.9208	0.9275	0.9208	0.9204
LSTM	0.4581	0.3050	0.4581	0.3039
RNN	0.5333	0.5484	0.5333	0.5304
SVM	0.9162	0.9366	0.9162	0.9160

Table B.2: ML Models Metrics for Rest vs Finger Spread

Metrics (Right) / ML Models (Right)	Accuracy	Precision	Recall	F1 Score
CNN	0.9200	0.9203	0.9200	0.9199
FNN	0.9445	0.9502	0.9445	0.9443
LSTM	0.4063	0.3693	0.4063	0.4585
RNN	0.6217	0.5032	0.6217	0.4999
SVM	0.8338	0.8396	0.8338	0.8343

Table B.3: ML Models Metrics for Rest vs Hand Wave In

Metrics (Right) / ML Models (Right)	Accuracy	Precision	Recall	F1 Score
CNN	0.9467	0.9473	0.9467	0.9440
FNN	0.9458	0.9589	0.9448	0.9441
LSTM	0.4705	0.3579	0.4705	0.3537
RNN	0.6419	0.6977	0.6419	0.6284
SVM	0.9448	0.9549	0.9448	0.9451

Table B.4: ML Models Metrics for Rest vs Hand Wave Out

Metrics (Right) / ML Models (Right)	Accuracy	Precision	Recall	F1 Score
CNN	0.9558	0.9550	0.9558	0.9539
FNN	0.9236	0.9376	0.9236	0.9220
LSTM	0.4438	0.3011	0.4438	0.2756
RNN	0.7690	0.5957	0.7690	0.6702
SVM	0.8526	0.8712	0.8526	0.8513

Table B.5: ML Models Metrics for Rest vs Double Finger Tap

Metrics (Right) / ML Models (Right)	Accuracy	Precision	Recall	F1 Score
CNN	0.9565	0.9649	0.9565	0.9581
FNN	0.9387	0.9610	0.9387	0.9431
LSTM	0.4036	0.3766	0.4036	0.4585
RNN	0.6929	0.5500	0.6929	0.5767
SVM	0.9304	0.9549	0.9304	0.9349

Table B.6: ML Models Metrics for Fist vs Finger Spread

Metrics (Right) / ML Models (Right)	Accuracy	Precision	Recall	F1 Score
CNN	0.9246	0.9158	0.9041	0.9187
FNN	0.9169	0.9253	0.9062	0.9147
LSTM	0.4690	0.4957	0.4491	0.4702
RNN	0.7690	0.5957	0.7387	0.6702
SVM	0.8214	0.8372	0.8114	0.8246

Table B.7: ML Models Metrics for Fist vs Hand Wave In

Metrics (Right) / ML Models (Right)	Accuracy	Precision	Recall	F1 Score
CNN	0.9565	0.9649	0.9473	0.9581
FNN	0.9206	0.9492	0.9136	0.9253
LSTM	0.4838	0.4734	0.4538	0.4577
RNN	0.7111	0.5589	0.6813	0.6064
SVM	0.9209	0.9467	0.9	0.9256

Table B.8: ML Models Metrics for Fist vs Hand Wave Out

Metrics (Right) / ML Models (Right)	Accuracy	Precision	Recall	F1 Score
CNN	0.9217	0.9229	0.9124	0.9207
FNN	0.9470	0.9577	0.9432	0.9481
LSTM	0.4221	0.4253	0.4571	0.5005
RNN	0.7111	0.5479	0.7022	0.6102
SVM	0.8771	0.8896	0.8273	0.8775

Table B.9: ML Models Metrics for Fist vs Double Finger Tap

Metrics (Right) / ML Models (Right)	Accuracy	Precision	Recall	F1 Score
CNN	0.9250	0.9297	0.9189	0.9235
FNN	0.9495	0.9592	0.9433	0.9493
LSTM	0.4163	0.3693	0.4133	0.4585
RNN	0.6319	0.5673	0.6334	0.5179
SVM	0.8035	0.8095	0.8123	0.8042

Table B.10: ML Models Metrics for Finger Spread vs Hand Wave In

Metrics (Right) / ML Models (Right)	Accuracy	Precision	Recall	F1 Score
CNN	0.9117	0.9043	0.9011	0.9045
FNN	0.9169	0.9253	0.9188	0.9147
LSTM	0.4690	0.3957	0.4567	0.3702
RNN	0.7690	0.5957	0.7523	0.6702
SVM	0.8018	0.8155	0.7944	0.8054

Table B.11: ML Models Metrics for Finger Spread vs Hand Wave Out

Metrics (Right) / ML Models (Right)	Accuracy	Precision	Recall	F1 Score
CNN	0.9498	0.9472	0.9501	0.9480
FNN	0.9171	0.9322	0.9073	0.9158
LSTM	0.4290	0.3757	0.4225	0.3702
RNN	0.7754	0.6449	0.7442	0.6840
SVM	0.8335	0.8415	0.8112	0.8295

Table B.12: ML Models Metrics for Finger Spread vs Double Finger Tap

Metrics (Right) / ML Models (Right)	Accuracy	Precision	Recall	F1 Score
CNN	0.9450	0.9456	0.9340	0.9448
FNN	0.9495	0.9578	0.9433	0.9496
LSTM	0.3967	0.3673	0.4088	0.2585
RNN	0.6317	0.5899	0.6113	0.5498
SVM	0.8538	0.8610	0.8433	0.8544

Table B.13: ML Models Metrics for Hand Wave In vs Hand Wave Out

Metrics (Right) / ML Models (Right)	Accuracy	Precision	Recall	F1 Score
CNN	0.9867	0.9886	0.9867	0.9868
FNN	0.9314	0.9469	0.9314	0.9315
LSTM	0.4438	0.3661	0.4438	0.3201
RNN	0.5457	0.5985	0.5457	0.4859
SVM	0.9448	0.9589	0.9448	0.9451

Table B.14: ML Models Metrics for Hand Wave In vs Double Finger Tap

Metrics (Right) / ML Models (Right)	Accuracy	Precision	Recall	F1 Score
CNN	0.9478	0.9531	0.9478	0.9493
FNN	0.9296	0.9545	0.9296	0.9342
LSTM	0.2838	0.1734	0.2838	0.1577
RNN	0.6933	0.5521	0.6933	0.5884
SVM	0.8949	0.9305	0.8949	0.8997

Table B.15: ML Models Metrics for Hand Wave Out vs Double Finger Tap

Metrics (Right) / ML Models (Right)	Accuracy	Precision	Recall	F1 Score
CNN	0.9304	0.9308	0.9304	0.9280
FNN	0.9561	0.9642	0.9561	0.9569
LSTM	0.2838	0.1734	0.2838	0.1577
RNN	0.6929	0.5500	0.6929	0.5767
SVM	0.8067	0.8140	0.8067	0.8065

Appendix C

3D SIMULATION RESULTS

Our simulation is compatible with a several platforms and includes adaptive performance features. Adaptive Performance in Unity for mixed reality games optimizes performance across various hardware configurations, ensuring a smooth and responsive user experience. It dynamically adjusts resolution, quality settings, and frame rates based on the device's capabilities and real-time performance metrics, while also managing thermal and power consumption to prevent overheating and excessive battery drain. This ensures that the simulation runs efficiently on devices such as Android, iOS, Windows, MacOS, and more, while maintaining high performance and user immersion.

The simulation supports Vulkan and DX11 (Direct3D11) Graphics API for Windows and sRGB for macOS. For Android, it uses Vulkan and OpenGL ES3, while for iOS, the simulation is set to use sRGB. Figures C.1 and C.2 illustrate the Game Flow scenes within the simulation. These screenshots alone do not portray specific task functionality, but rather are meant to showcase different visuals in the simulation. In these scenes, players use Myo arm-bands to execute hand movements according to the two *game flows* $\mathbb{G}_1$ and $\mathbb{G}_2$. Depending on the hand movements, the character in the simulation performs a specific set of actions and animations.

Figure C.3 shows a feature where players can create custom game flows based on their rehabilitation needs. This part of the simulation is not completed at this point and its development is planned for future work.

Profiler

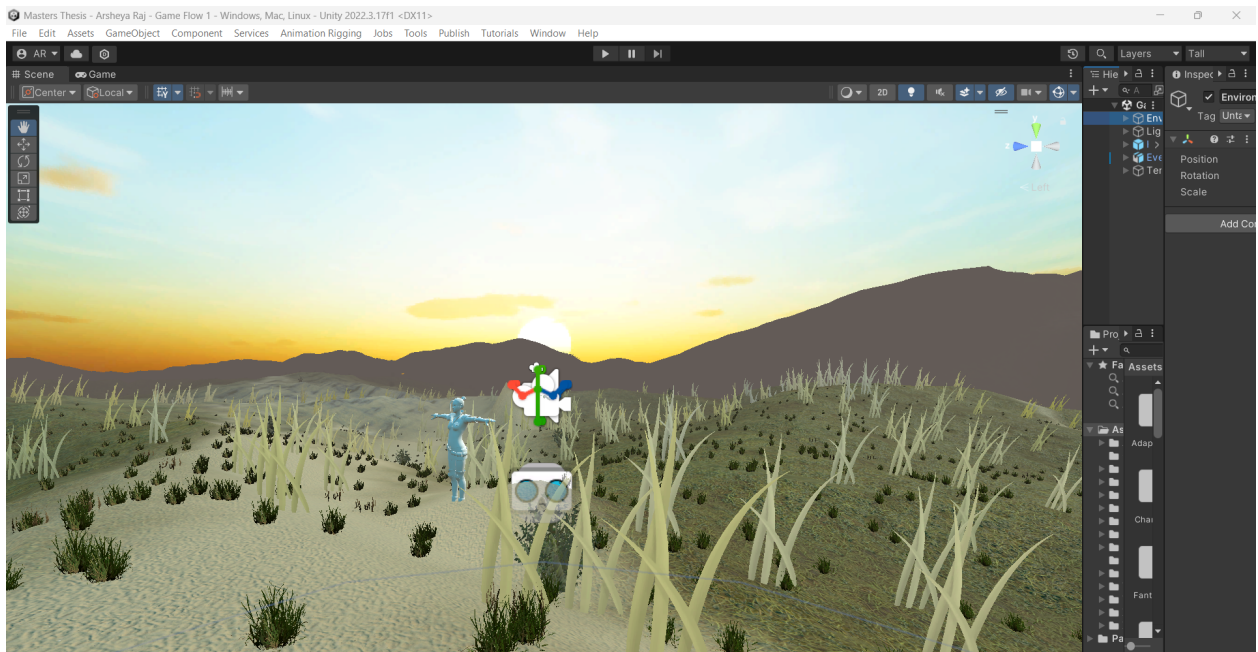


Figure C.1: Game Flow 1 scene in Unity

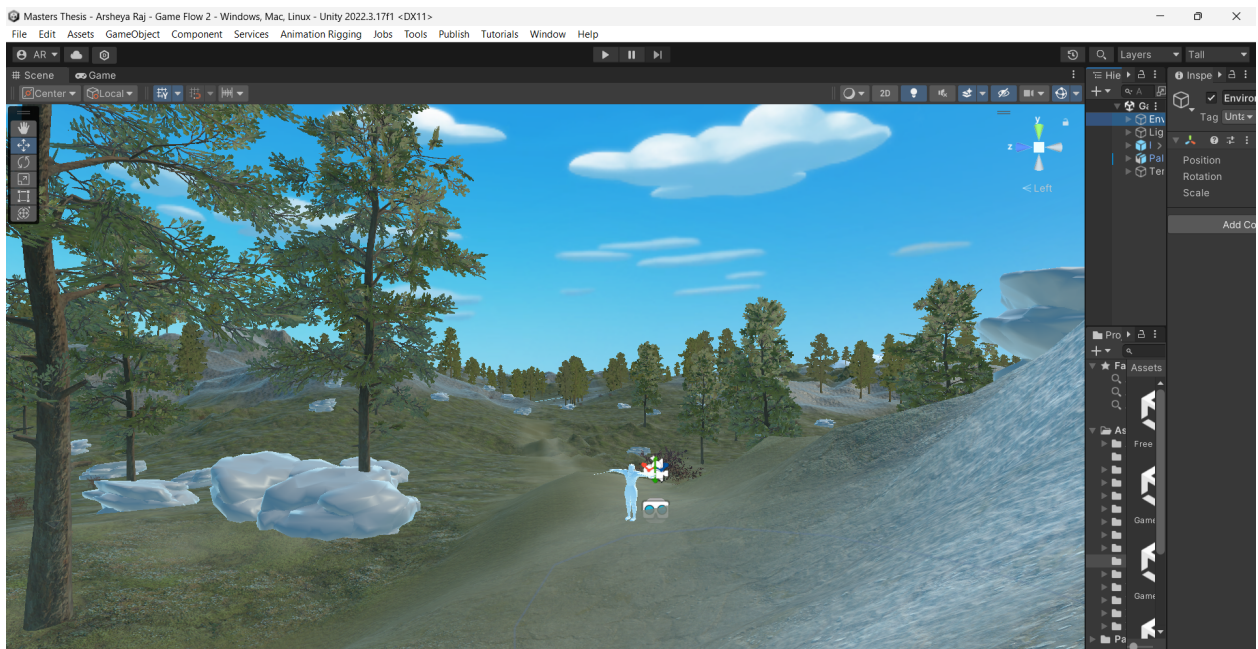


Figure C.2: Game Flow 2 scene in Unity

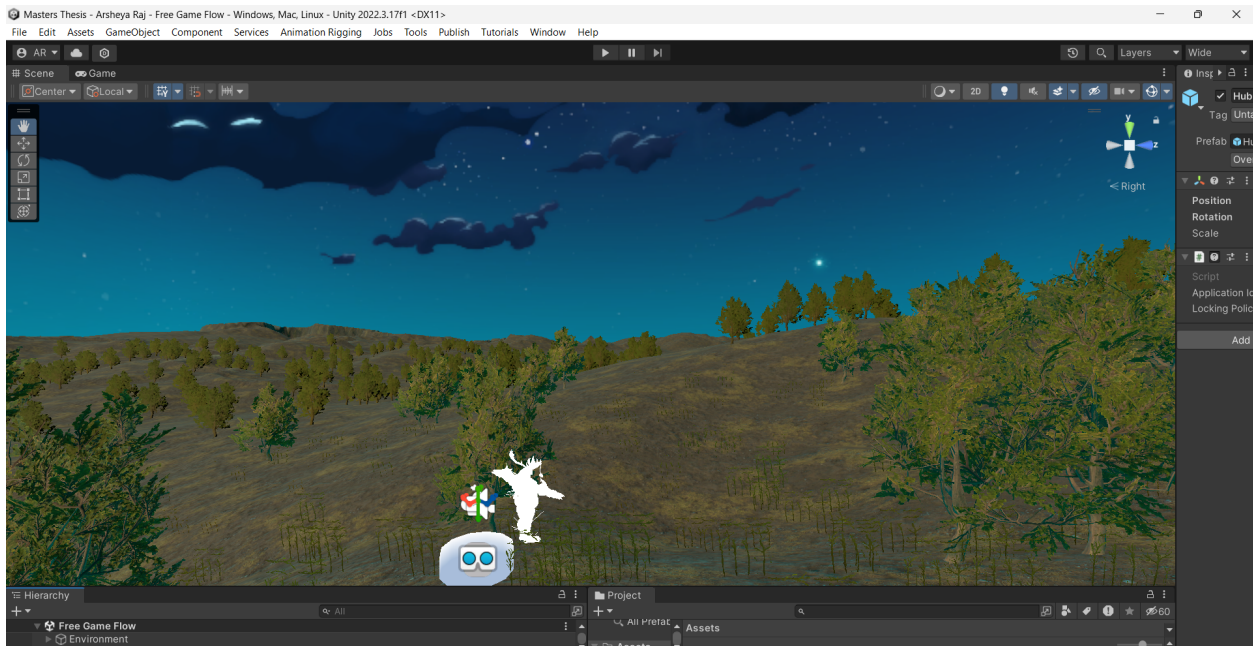


Figure C.3: Free Game Flow scene in Unity

Figure C.4 displays the results of the game profiler for the DX11 Graphics API.

The metrics and methods for the quality assessment of the game system are as follows :

- **Frame Rate and Latency:** In AR/MR, frame rate (frames per second) and latency (delay in response time) are crucial for a smooth and comfortable user experience. Low frame rates and high latency can lead to motion sickness and discomfort.
- **CPU/GPU Usage:** Monitors the amount of processing power used by the game. Efficient usage ensures smooth performance.
- **Comfort and Immersion:** Evaluates how comfortable and engaging the experience is, often assessed through user feedback.
- **Session Duration:** Average time users spend in the game.
- **Gesture recognition accuracy:** Gesture recognition accuracy with the Myo Armband measures how correctly the system identifies and classifies gestures performed by users, typically ranging from 80% to 90% depending on factors like gesture complexity and

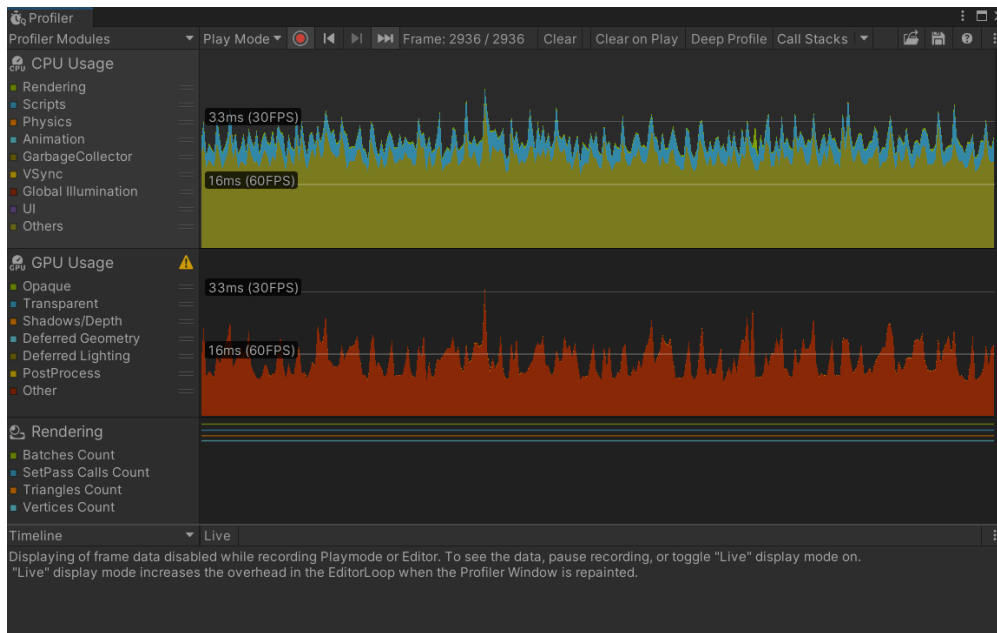


Figure C.4: Unity Profiler: CPU and GPU usage

data quality. Achieving high accuracy involves optimizing data collection, preprocessing, feature extraction, and model training.

During testing, the game ran smoothly without performance issues, but it was imperative to understand and assess related metrics to understand how the simulation performed. Relevant performance-related metrics considered are:

CPU Usage: The CPU usage of approximately 33 ms per frame, relative to the target of 30 FPS (frames per second), suggests that the application is slightly exceeding the desired frame time. Consistent frame rates are ideal for ensuring smooth gameplay; if the CPU usage remains stable and does not experience significant spikes, it may still be acceptable. However, if specific categories such as Scripts or Garbage Collector consistently consume excessive CPU time, it would be prudent to focus on optimization efforts.

GPU Usage: The GPU usage of approximately 1.6 ms per frame, against the target of 60 FPS, indicates efficient rendering. This value aligns well with the desired frame rate,

suggesting that the graphics pipeline is performing optimally. It is important for developers to continue monitoring GPU usage to ensure a proper balance between visual quality and performance.

In summary: CPU usage is a bit higher than ideal but still manageable, while GPU usage is within the expected range. The increased CPU usage was partly due to other applications running at the same time, but it's still acceptable.

Appendix D

MYO CONNECT - MYO ARMBAND MANGER

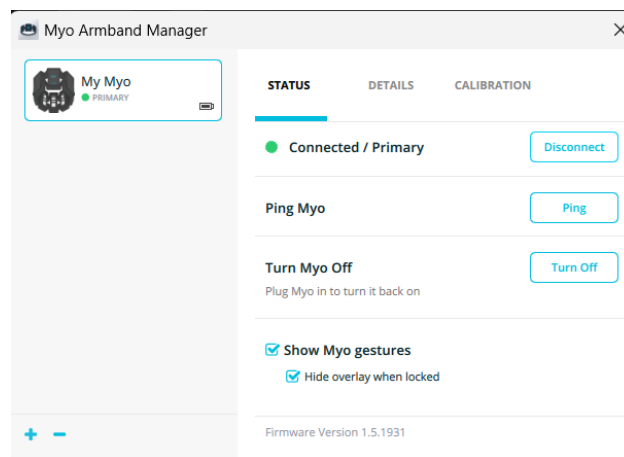


Figure D.1: Myo Connect-Armband Manager

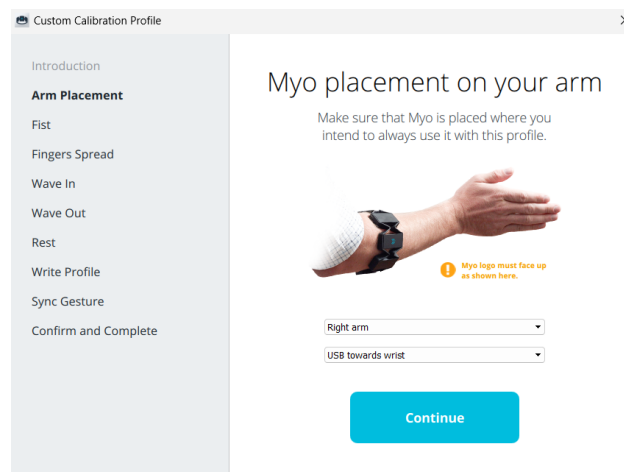


Figure D.2: Myo Connect-Custom Profile Calibration

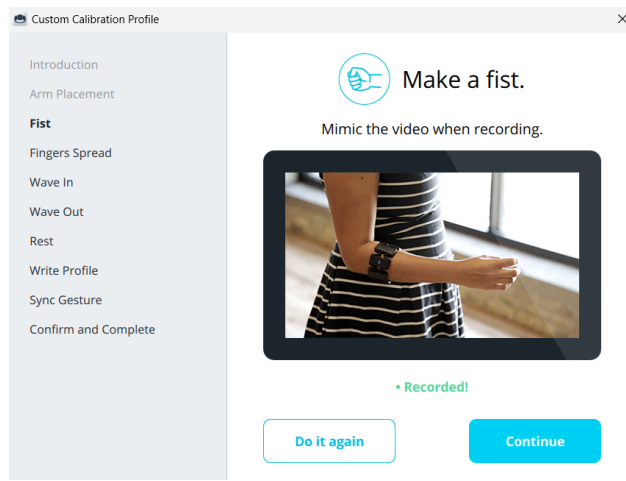


Figure D.3: Myo Connect-Make a fist

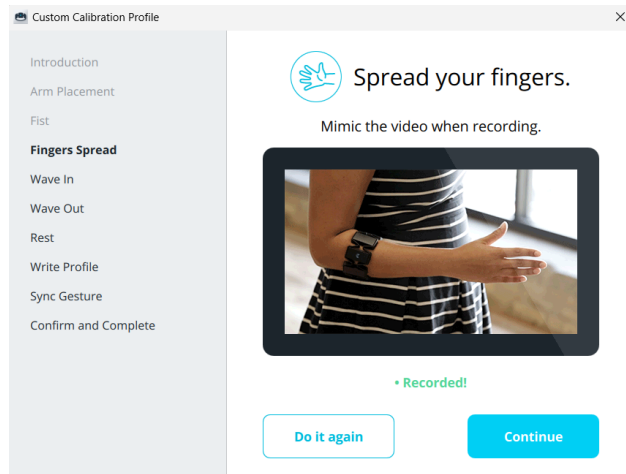


Figure D.4: Myo Connect-Spread your fingers

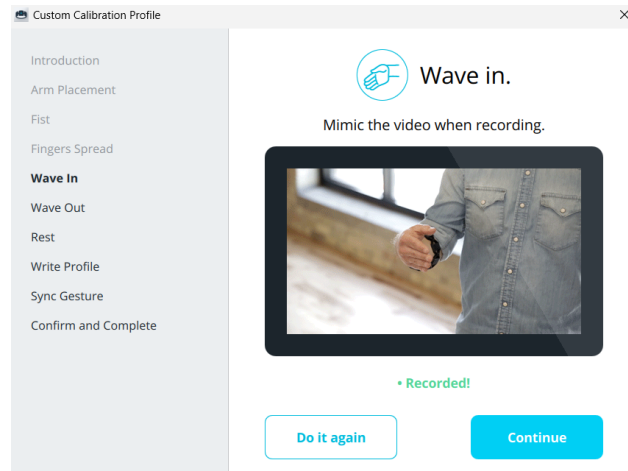


Figure D.5: Myo Connect-Wave In

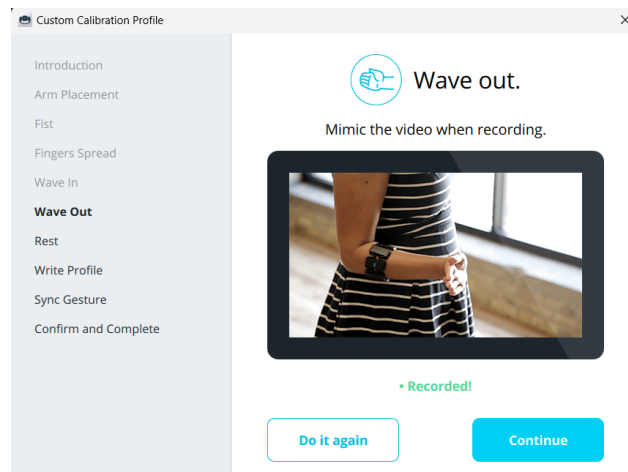


Figure D.6: Myo Connect-Wave Out

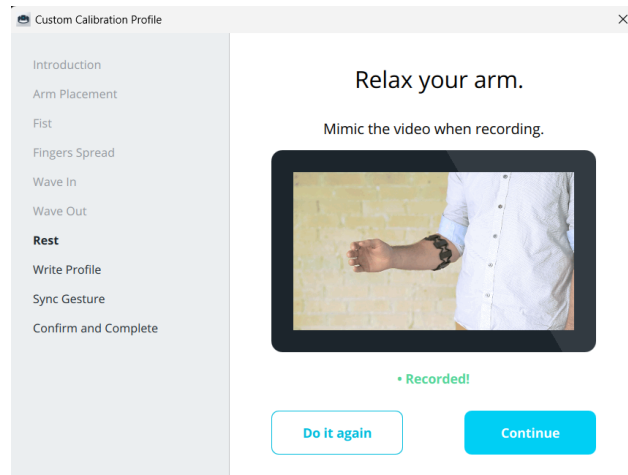


Figure D.7: Myo Connect-Relax your arm

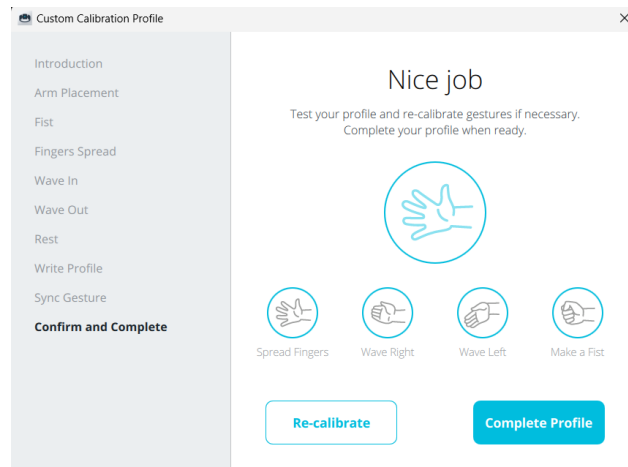


Figure D.8: Myo Connect-Profile Calibration Complete.png

Appendix E

DSI STREAMER

E.1 Panels

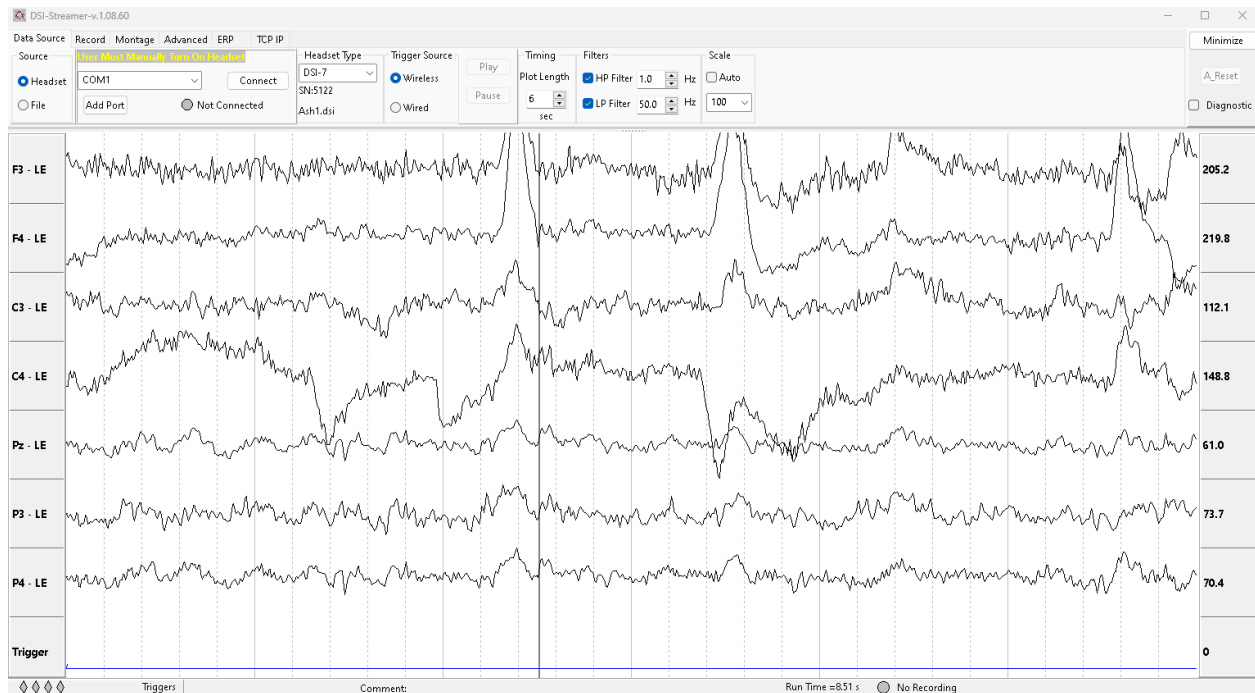


Figure E.1: Data Source Panel in Streamer. Used to connect to DSI7 EEG Headset.

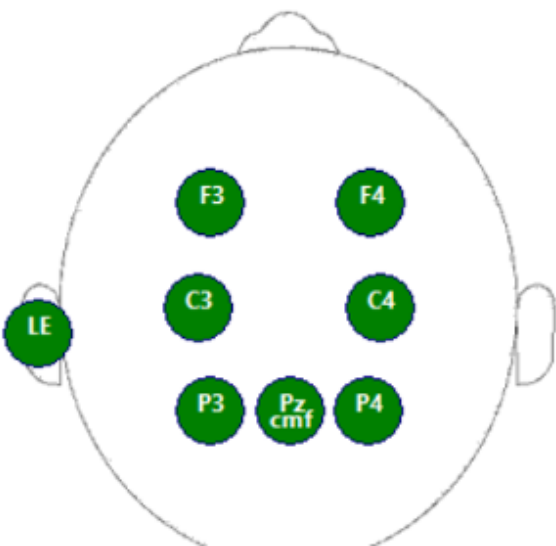


Figure E.2: Montage Panel in DSI Streamer. Used to calibrate each electrode on the DSI7 EEG Headset.

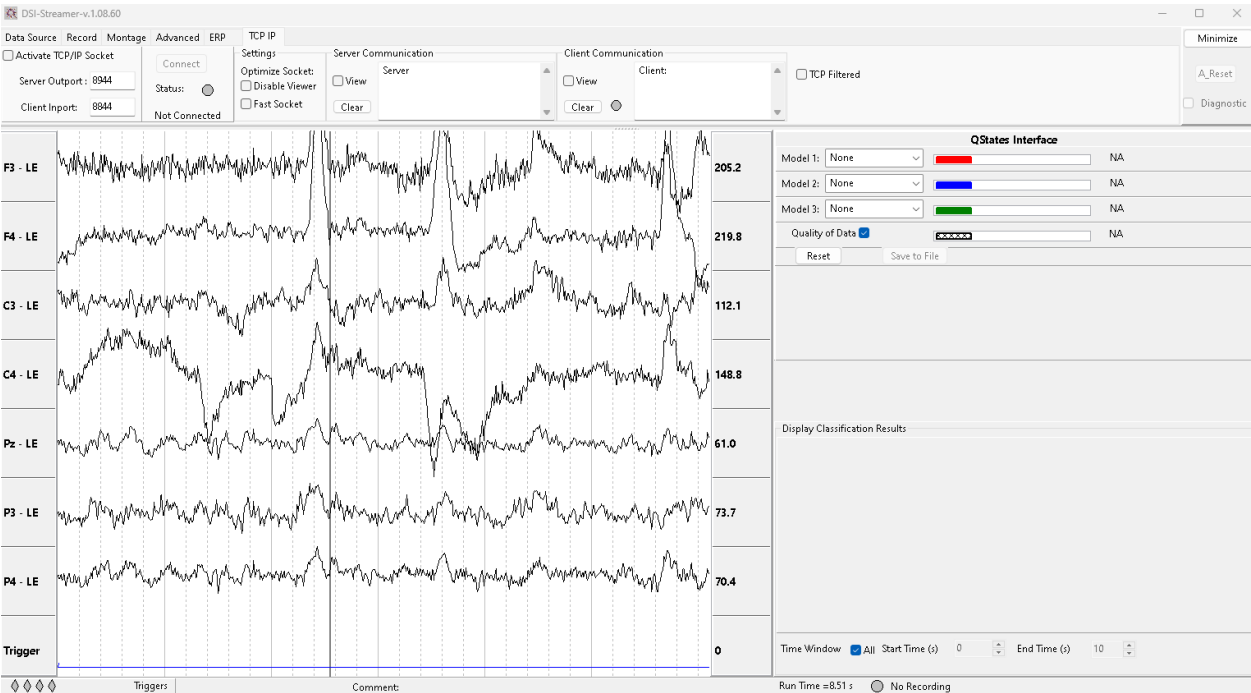


Figure E.3: TCP/IP Panel in DSI Streamer.

E.2 Known Bugs

E.2.1 TCP/IP Program Slowdown/Speedup

During data collection a critical bug was discovered in DSI Streamer. This bug involved the order in which data collection and the TCP/IP Socket was setup. It is considered critical to this work because it negatively impacted data collection and resulted in slowdown of the program.

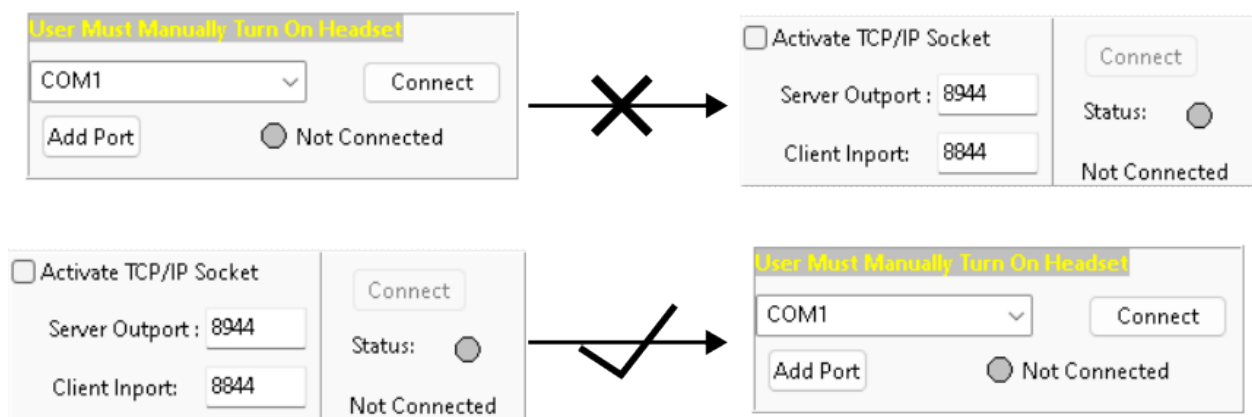


Figure E.4: How to avoid the TCP/IP Slowdown Bug. Do not disable TCP/IP during Data collection either.

Cause In DSI Streamer, connecting the headset or replaying a file before opening the TCP/IP socket leads to an immediate slowdown of the program. This slowdown not only affects the timing but also reduces the rate at which data is transmitted over the TCP/IP socket.

Remedy Open the TCP/IP socket before connecting the headset or replaying a file. If the bug has already occurred, disabling and then re-enabling the TCP/IP socket can also resolve the issue.

E.2.2 Port Connection for the DSI Streamer

Additionally, using the Bluetooth adapter of the Thalmic Labs Myo Armband prevented the DSI Streamer from connecting to the headset via COM3 port.

Cause The Bluetooth Adapter of the Myo armband was by default connected through the port of COM3.

Remedy To resolve the COM3 port bug, create a DSI_Port_history.lst file and assign a different port, such as COM4, to the DSI streamer.

E.2.3 DSI Streamer Version

Another issue discovered was that the Dry DSI-7 Electrode Headset was only compatible with DSI-Streamer-v.1.08.60, while the updated Wearable Sensing DSI Streamer only worked with the DSI-24 Electrode Headset.