Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

Helene Tørlen Lønvik
Pauline Mørch Jonassen

An Exploration of Techniques for Electroencephalography-Based Motor Imagery Classification for Real-Time Drone Control

Master's thesis in Cybernetics and Robotics
Supervisor: Marta Molinas
Co-supervisor: Luis Alfredo Moctezuma
May 2023

NTNU
Norwegian University of
Science and Technology

Helene Tørlen Lønvik
Pauline Mørch Jonassen

An Exploration of Techniques for Electroencephalography-Based Motor Imagery Classification for Real-Time Drone Control

Master's thesis in Cybernetics and Robotics
Supervisor: Marta Molinas
Co-supervisor: Luis Alfredo Moctezuma
May 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

NTNU
Norwegian University of
Science and Technology

Preface

This Master's thesis completes a Master of Technology at the Norwegian University of Science and Technology (NTNU) under the Department of Engineering Cybernetics. It was carried out during the spring semester of 2023, building on a specialization project written by the two authors in the fall semester of 2022. Both of the authors have been working together on all the aspects of this thesis, meaning that the project has not been split in any way.

The idea behind the thesis was provided by Marta Molinas, who also acted as our supervisor for both the specialization project and the thesis. In addition to being our supervisor, Molinas provided the code for EEGnet written by Vernon J. Lawhern et al., and the dataset collected by Steryl et al.

Luis Alfredo Moctezuma acted as our co-supervisor for this thesis. He suggested and provided code examples for Discrete Wavelet Transform (DWT) and Non-dominated Sorting Genetic Algorithm (NSGA), in addition to helping with other code implementations and finding relevant sources. The other methods implemented and used throughout this thesis were found in various sources and through a literature search carried out by us, with the support and suggestions from our supervisors.

We had no prior knowledge about Motor Imagery (MI), Electroencephalography (EEG), or NSGA when starting the specialization project in the fall of 2022. In addition, we had limited experience in signal processing, decomposition methods, Deep Learning (DL), and Transfer Learning (TL). Hopefully, the new knowledge and skills gained through this project will be reflected during the thesis. A video of an online test of the drone actuation can be found here (https://youtu.be/SVvtyDZMIqg).

Trondheim, 2023-05-26
Pauline Mørch Jonasen and Helene Tørlen Lønvik

Acknowledgement

We would like to express our sincere gratitude to our supervisor Marta Molinas for giving us the opportunity to do our research in this interesting field of study, and for giving us her insights and guidance throughout the work of this thesis. We will also express our gratitude to our co-supervisor Luis Alfredo Moctezuma for being available and providing help and guidance whenever it was needed. Further, we want to thank both of them for listening to and discussing our ideas and having faith in our work and our abilities.

Further, we would like to thank the other master students sitting at our master's office for being cheerful and helpful fellow students. Thanks for listening to our thoughts when we needed to think out loud and contributing critical and uplifting feedback.

Lastly, we would also like to thank each other for the collaboration on this thesis, and for motivating and cheering each other on. We would also like to thank our family and friends for supporting us and cheering us on through this thesis.

Abstract

This thesis explores different state-of-the-art techniques for preprocessing, feature extraction, and classification of Electroencephalography (EEG) Motor Imagery (MI) data. The goal was to develop a Brain-Computer Interface (BCI) that can accurately classify different MI tasks and interpret them into commands for controlling a drone in real-time.

Accurately classifying MI poses a significant challenge, and remains an active area of research. As the field of study is progressing, numerous approaches have been suggested. Based on a literature review, a selection of approaches were chosen and further exploited using a Multi-objective Evolutionary Algorithm called the Non-dominated Sorting Genetic Algorithm (NSGA). The NSGA algorithm was used for optimizing subject-dependent methods. Further, Transfer Learning (TL) was used to exploit the possibility of making subject-independent models.

After initial testing on the dataset consisting of 14 subjects, a subset of 4 subjects was selected. For the offline classification, models for both 2 and 3 classes were implemented. For the 2-class classification (right-hand and foot), subjects S03 and S09 yielded the highest accuracies. Subject S03 obtained an accuracy of 100.00% with various Machine Learning (ML) methods and with the Deep Learning (DL) model EEGnet. Subject S09 achieved accuracies as high as 93.80% with EEGnet and 87.50% with Common Spatial Pattern (CSP) feature extraction and Random Forest (RF) classification. For three-class classification (right-hand, foot, and rest), subject S03 obtained accuracies of 100.00% and 90.00% with EEGnet and CSP-RF, respectively. Subject S09 obtained 87.50% and 65.00% with EEGnet and CSP-RF, respectively. All results were obtained using a flat model structure. Extensive testing of different TL models showed no significant increase in accuracy overall, except for subject S09, which achieved an accuracy of 95.00%.

For the online classification and drone actuation with two-class classification, subject S03 obtained an accuracy of 81.67% with EEGnet and 89.17% with CSP-RF. Subject S09's best performance led to an accuracy of 84.17% using EEGnet and 82.50% using CSP-RF. When introducing the rest class, the accuracy decreased for both subjects. The highest obtained accuracy for subject S03 was 77.27% and 72.27% for EEGnet and CSP-RF respectively. For subject S09 it was 74.89% and 67.12% for EEGnet and CSP-RF, respectively.

Sammendrag

Denne masteroppgaven undersøker ulike state-of-the-art teknikker for preprosessering, egenskapsekstraksjon og klassifiserings av Electroencephalography (EEG) Motorisk Innbilning (MI) data. Målet for oppgaven var å utvikle et hjerne-datamaskin-grensesnitt (BCI) som nøyaktig kunne klassifisere ulike motoriske innbilninger og tolke dem som kommandoer for å styre en drone i sanntid.

Nøyaktig klassifisering av MI-oppgaver utgjør en betydelig utfordring og forblir et aktivt forskningsområde. Ettersom forskningsfeltet utvikler seg, er det også blitt foreslått en rekke tilnærminger og metoder. Basert på en litteraturgjennomgang ble det valgt ut flere metoder som deretter ble utforsket videre ved hjelp av en flerobjektiv evolusjonsalgoritme kalt Non-Dominated Sorting Genetic Algorithm (NSGA). NSGA-algoritmen ble brukt til å optimalisere modeller som var subjekt-avhengige. Videre ble overføringslæring (TL) benyttet for å lage modeller uavhengig av subjektet.

Etter innledende testing på datasettet bestående av 14 subjekter, ble en undergruppe på 4 subjekter valgt. For offline-klassifiseringen ble modeller for både to og tre klasser implementert. For klassifiseringen med to klasser (høyrehånd og fot), var de høyeste nøyaktighetene oppnådd med subjektene S03 og S09. Subjekt S03 oppnådde en nøyaktighet på 100.00% med ulike maskinlæringsmetoder (ML) og med Dyp Læringsmodellen (DL) EEGnet. Subjekt S09 oppnådde nøyaktigheter på 93.80% med EEGnet og 87.50% med Common Spatial Pattern (CSP) egenskapekstraksjon og Random Forest (RF) klassifisering. For klassifisering med tre klasser (høyrehånd, fot og hvile), oppnådde subjekt S03 nøyaktigheter på 100.00% og 90.00% med henholdsvis EEGnet og CSP-RF. Subjekt S09 oppnådde 87.50% og 65.00% med henholdsvis EEGnet og CSP-RF. Alle resultater ble oppnådd ved bruk av en flat modellstruktur. Omfattende testing av forskjellige TL-modeller viste ingen markant økning i nøyaktighet totalt sett, unntatt for subjekt S09, som oppnådde en nøyaktighet på 95.00%.

For online-klassifiseringen og droneaktivering med klassifisering av to klasser, oppnådde subjekt S03 en nøyaktighet på 81.67% med EEGnet og 89.17% med CSP-RF. Subjekt S09s prestasjon førte til en nøyaktighet på 84.17% ved bruk av EEGnet og 82.50% ved bruk av CSP-RF. Ved introduksjon av hvileklassen ble nøyaktigheten redusert for begge subjektene. Den høyeste oppnådde nøyaktigheten for subjekt S03 var 77.27% og 72.27% for henholdsvis EEGnet og CSP-RF, mens for subjekt S09 var den 74.89% og 67.12% for henholdsvis EEGnet og CSP-RF.

Contents

List of Figures

List of Tables

List of Abbreviations

API Application Programming Interface

BCI Brain-Computer Interface

CAR Common Average Rereferencing

CNN Convolutional Neural Network

CSP Common Spatial Pattern

DFT Discrete Fourier Transform

DL Deep Learning

DNN Deep Neural Network

DWT Discrete Wavelet Transform

EEG Electroencephalography

ELM Extreme Learning Machine

ERD Even-Related Desynchronization

ERS Event-Related Synchronization

fbCSP filter bank Common Spatial Pattern

FPR False Positive Rate

GA Genetic Algorithm

GB Gradient Boosting

HFD Higuchi Fractal Dimension

HMO Hjort Mobility

HCO Hjort Complexity

IE Instantaneous Energy

KFD Katz Fractal Dimension

LDA Linear Discriminant Analysis

LL Line Length

LSTM Long Short-Term Memory

ML Machine Learning

MLP Multi-layer perceptron

MI Motor Imagery

NN Neural Network

NSGA Non-dominated Sorting Genetic Algorithm

PFD Petrosian Fractal Dimension

PTP Peak-To-Peak

RF Random Forest

RNN Recurrent Neural Network

RMS Root Mean Square

SD Standard Deviation

SNR Signal-to-Noise Ratio

SMR Sensorimotor Rhythms

sLDA Shrinkage Linear Discriminant Analysis

SVM Support Vector Machine

SFD Sevick Fractal Dimension

TE Teager Energy

TL Transfer Learning

TPR True Positive Rate

TNR True Negative Rate

Chapter 1

Introduction

1.1 Background

The human brain is composed of billions of interconnected nerve cells called neurons, and when activated, these neurons send out signals that can be recorded using Electroencephalography (EEG). To register these signals, electrodes that are connected to a Brain-Computer Interface (BCI) is used. BCIs are computer systems that provide the possibility of interaction between the brain and external devices and can be worn externally like a helmet or implanted into the brain.

MI is a neuroparadigm that can be recorded using a BCI. It occurs when a person imagines a movement, making the field of MI research interesting for applications that can be used as neurorehabilitation, neuroprosthetics, and assistants for the physically impaired. In addition to being a supplement in the health sector, it can provide value in applications like gaming, monitoring, and drone control.

As MI and other EEG signals are often noisy and prone to artifacts, making a good classification model can be difficult. Furthermore, brain activity from other motor functions occurs in the same part of the brain as MI signals, affecting how accurately the model can be built. These problems reduce the accuracy, robustness, efficiency, and consistency of the MI-BCIs, making it important to develop better sensors and improve signal processing techniques and classification methods for MI.

Further, as MI is a learnable trait, the ability to perform MI can vary a lot from subject to subject. In addition, the occurrence of MI signals will likely not be at the same specific location of the brain, and the brain waves may not look exactly the same for each subject. This implies that different features and classifiers might work better for some subjects than others, making it hard to find a subject-independent model.

Live classification of MI signals can be performed using synchronous or asynchronous BCIs. Synchronous BCIs, also named reactive BCIs, receive commands in specific time intervals, often after a given cue. Asynchronous BCIs, also known as self-paced BCIs, make it possible to control a device at a subject's own pace. Instead of detecting specific brain signals or patterns in a specific time interval, it continuously monitors brain activity, letting the subject interact whenever it wants. This makes a BCI more flexible. On the other hand, asynchronous BCIs have typically performed worse than synchronous BCIs, making it important to improve this paradigm for better communication between the user and a device.

1.2 Motivation

Based on the background, a big motivation for writing this master's thesis stems from the potential of conducting research on MI-BCIs which can help to improve the quality of individuals with motor impairments. Moreover, such research can contribute to improving the knowledge in the field of neuroprosthetics, making it possible to develop devices that can restore lost motor functions. Additionally, MI-BCIs can be used in new and exciting fields where humans can interact with computers and digital devices, exemplified by the focus of this thesis which is the actuation of a drone. Lastly, a great motivation for carrying out this thesis is that conducting research on brain activity will advance our understanding of the brain and how it processes information.

1.3 Objective

This thesis aims to design and implement a model to actuate and control a drone using MI-signals. As a part of this work, a robust Machine Learning (ML) or Deep Learning (DL) model must be designed, to ensure that the drone avoids getting uncontrollable. The system should be tested in real-time, using a stream of data.

The problem of implementing a BCI for controlling a drone using MI signals can be broken down into the following main objectives:

1. Research state-of-the-art algorithms for signal processing, decomposition and classification of MI signals.
2. Implement and test different algorithms and pipelines, aiming to increase the accuracy to its highest.
3. Find the pipelines working best for both for classifying different MI tasks, but also separating between MI and rest state.
4. Design and implement a state machine for simple control of a drone using the best pipeline found from the testing.
5. Make it possible to operate the drone with a stream of data.

1.4 Approach

The objective of this thesis was approached in the following matter. First, a literature review was conducted to survey the existing work performed concerning preprocessing, feature extraction, and classification methods in the field of MI. Further, data from a public dataset was used for experimentation and optimization of pipelines. The dataset consists of 14 subjects and after some initial testing, a subset of subjects was chosen for further experimentation.

The different pipelines tested were based on feature extraction and classification. For feature extraction the 3 different methods Common Spatial Pattern (CSP), Discrete Wavelet Transform (DWT), and bandpass-based feature extraction were tested, while for classification, the methods used were Random Forest (RF), Linear Discriminant Analysis (LDA), Support Vector Machine (SVM), Gradient Boosting (GB), and EEGnet. Furthermore, the preprocessing techniques Common Average Rereferencing (CAR) and Laplace Rereferencing were implemented to see if they could enhance performance.

To find the best pipelines, electrodes, and features, an optimization was done using the popular non-domination-based genetic algorithm for multi-objective optimization, Non-dominated Sorting Genetic Algorithm (NSGA). The evaluation of the different methods employed in this study was primarily centered on the accuracy score, which served as the primary metric for assessing the performance of the predicted MI classes.

In addition to optimizing the pipelines, Transfer Learning (TL) was tested to see if transferring knowledge from a subset of subjects to another could make good classification models. This was done for both the DL and ML pipelines.

To be able to fly a drone in real-time, a time window optimization was conducted to find the optimal 1-second time window most suitable for the model. In addition, state machines were produced to decide how the drone should move based on the classification done from the model.

1.5 Limitations

The primary objective of this thesis revolves around testing and optimizing different methods for signal processing, feature extraction, classification, and electrode selection. These experiments are quite time-consuming, resulting in constraints on the time used for live implementation and testing. Consequently, the time allocated for conducting experiments involving drone actuation was shorter than originally desired. Furthermore, due to time limitations, data collection and the construction of models enabling real-time actuation of the drone instead of relying on a data stream were not feasible.

1.6 Outline

The structure of this thesis is as follows: Chapter 2 presents relevant theory for the thesis, giving an explanation of brain functions, and different methods used for signal processing, feature extraction, and classification. The theory gives a foundation for the experimental results and analysis in the subsequent chapters. Further, Chapter 3 presents a state-of-the-art literature review of research done in the field of MI classification. In addition, it presents a review of results obtained from other studies of the dataset used in this thesis. Chapter 4 includes a description of the dataset, tools, procedures for data acquisition, and methods used in the experiments. The experiments and their results are presented in Chapter 5 before they are discussed in Chapter 6. Chapter 6 also includes a conclusion and recommendation for further work.

Chapter 2

Theoretical Background

2.1 Signals From The Human Brain

2.1.1 Brain Signals and Frequency Bands of Brain Signals

The human brain is a complex system that is an interconnection of billions of nerve cells called neurons. These neurons communicate and transfer information by employing and encoding different types of electrical signals[1]. The oscillations of the electrical signal in the brain can be divided into five different frequency bands. The bands are delta (δ), theta (θ), alpha (α) also called mu rhythm, beta (β), and gamma (γ) [2]. The bands and their respective frequency range can be seen in table 2.1.

Table 2.1: Frequency bands of brain signals[3].

Frequency band	Frequencies
delta (δ)	$< 4Hz$
theta (θ)	$4Hz - 8Hz$
alpha (α) / mu	$8Hz - 13Hz$
beta (β)	$13Hz - 30Hz$
gamma (γ)	$> 31Hz$

2.1.2 Motor Cortex

The human brain is divided into four more significant regions, the frontal lobe, parietal lobe, occipital lobe, and temporal lobe[2]. The sectioning of the brain is shown in figure 2.1. The frontal lobe can further be divided into sections with several larger regions that are directly involved in the execution of different motor tasks[4]. The largest part of the frontal lobe is called the primary motor cortex. The primary motor cortex includes, among other things, the movement of feet, arms, and head. The location of the primary motor cortex can be seen in figure 2.1.

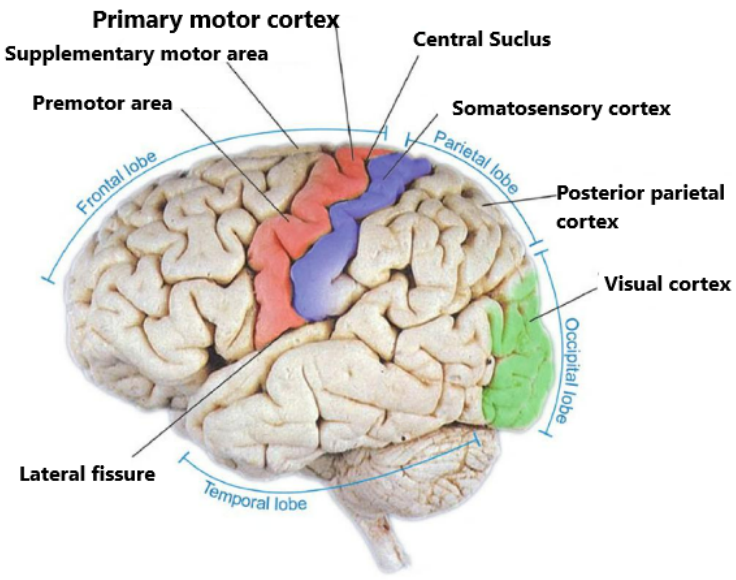


Figure 2.1: The human brain and its significant regions, frontal, parietal, occipital, and temporal lobe and primary motor cortex. Illustration adapted from [2].

The locations which control different body parts can be seen in figure 2.2. From this figure, it can be seen that the size controlling different body parts is variable and that the hand takes up much more area than the feet. The variable sizes are dependent on how much precision the different body parts require. The illustration shows that the location of the feet is more medial, and the hands more lateral[4].

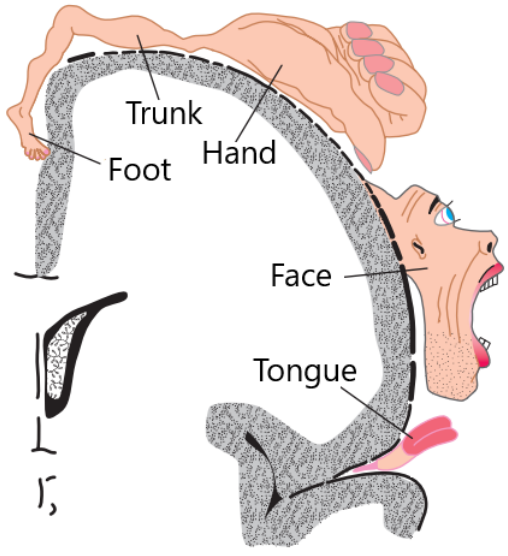


Figure 2.2: Primary motor cortex sectioned into the respective parts of the body. The feet most medially, hands more laterally, and face and tongue the most lateral. Illustration adapted from[4].

2.2 Electroencephalography

EEG is a non-invasive method for examining the human brain, where the signals are sampled by placing electrodes on the scalp. The electrodes measure the direct cortical activity, which is the changes in amplitudes of the electrical impulses. For a typical adult, the amplitudes are measured to range between $10\mu V - 100\mu V$[3]. The EEG signals are usually non-linear, non-Gaussian, random, and non-correlated, and can be used to extract features from brain signals. The features extracted by the use of EEG are dependent on several factors, among them the subject itself, age, and mental health [3]. EEG is commonly used for detecting injuries or disorders in the brain, including epilepsy, sleeping disorder, depression, tumor, stress, and trauma-related disorders[3].

2.2.1 Electrode Placement

To ensure standardization of electrode placements, a 10-20 electrode placing scheme is often used. The use of this standardized system ensures the possibility to reproduce a recording scheme and makes it possible to compare different studies. The 10-20 system is based on the location of the electrodes and the underlying cerebral cortex, where the 10-20 refers to the distance to adjacent electrodes. The distance is measured to be either 10% or 20% of the front-back or right-left distance of the skull[5]. The 10-20 placing scheme is illustrated in figure 2.3.

An extension of the 10-20 placing scheme is the 10-10 placing scheme. The 10-10 scheme is based on, and similar to the 10-20 scheme, but includes more electrodes in between the ones in the 10-20 system. More specifically, it fills electrodes halfway in between the 10-20 electrodes, which makes the distance between adjacent electrodes 10% of the front-back and right-left length of the skull[5]. The 10-10 placing scheme is illustrated in figure 2.4.

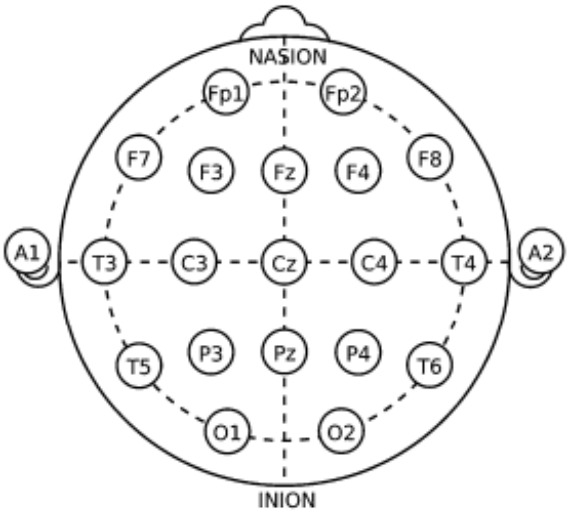


Figure 2.3: Illustration of the 10-20 placing scheme, where the distance between the adjacent electrodes is 10% or 20% of the front-back and left-right length of the skull. The illustration is reprinted from[5].

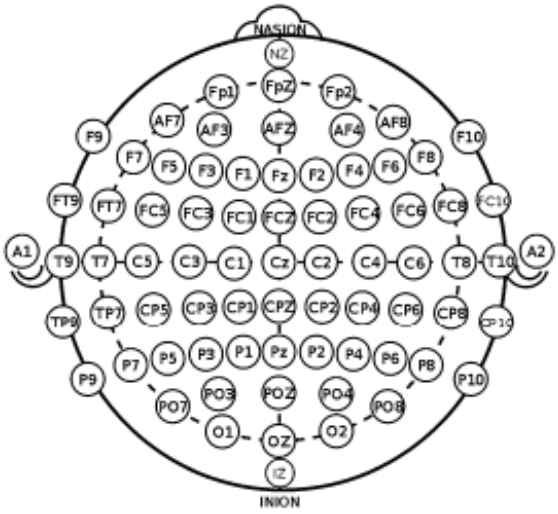


Figure 2.4: Illustration of the 10-10 placing scheme, where the distance between the adjacent electrodes is 10% of the front-back and left-right length of the skull. The illustration is reprinted from[5].

2.2.2 Artifacts in Electroencephalography

Artifacts in EEG can be divided into two categories: physiological artifacts, and non-physiological artifacts. The physiological artifacts originate from the body itself, and the most common artifact of this type is eye blinks, eye movements, head movements, heartbeats, and muscular noise. The non-physiological artifacts originate from the environments around the subject[5].

Physiological Artifacts

Ocular Artifacts also known as eye movement and eye blinks can be easily detected in EEG for conscious subjects, as the movement will give rise to a large electrical potential. This electrical potential might often be characterized by anterior location, bilateral and synchronized appearance, and is linearly summed upon the EEG signal[5]. Using extra electrodes above and below the eye can help to identify the ocular artifacts. There are also different decomposition and regression methods suggested for removing these artifacts and still preserving the information of the EEG signal[5].

Electromyography artifacts are often caused by muscular movements, these can be identified as high-frequency and spiky activities. The muscular movements usually involved are located in the frontalis and temporalis muscles, whereas activity in the frontalis is usually involved in forced eye closure and photic simulation. Jaw clenching, chewing, and bruxism involve movement in the temporalis muscles [5]. Contraction of the frontalis muscles might appear as a "railroad track", while temporalis contraction might appear as a burst of activities[5]. To avoid and/or minimize the effect of these types of artifacts, the sub-

ject can get instructions to keep the mouth open and try to focus the look in one place, i.e. at a fixation cross on a screen.

Electrocardiographic artifacts are artifacts that originate from the heart. These artifacts occur more often for subjects that are overweight, have short necks, and for babies. This is due to the fact that the dipole is located closer to the recording electrodes and is able to transmit the current to a higher level. To reduce this artifact, one linked ear montage can be used [5].

Perspiration artifact appears as very low-frequency, about $0.5Hz$, and very low-amplitude waves. This type of artifact arises due to unwanted electrical connections between the electrode on the scalp. Movements of the subject might also cause the electrodes to move and cause artifacts[5]. Therefore, giving the subject guidelines for keeping the movement to a minimum can remove or reduce this type of artifact.

Non-physiological Artifacts

Interference from the power supply is one of the most common non-physical artifacts, which in Europe is at $50Hz$. For reducing this artifact, shielding of cables and shielded recording rooms can be used. Another way to remove this artifact is to apply a notch filter for removing the $50Hz$ frequency[5].

Poor electrode placement might be the origin of another non-physiological artifact. Movements of an electrode might cause a "pop", which means a sudden positive discharge and can be characterized as an initial high-voltage very steep deflection, followed by an exponential decay[5]. This problem can be avoided by proper placement and cleaning and maintenance of the equipment[5].

2.3 Motor Imagery

MI can be described as the result of conscious access to the content of the intention of a movement, which is usually performed unconsciously during movement preparation[6]. In other words, MI is a mental process where a movement is imagined or prepared without any muscular activity. By comparing the conscious performance of MI and the unconscious preparation of an actually performed movement, they will contain many of the same mechanisms and will be almost equal[6]. The oscillations in brain activity that are recorded from the somatosensory and motor areas are known as Sensorimotor Rhythms (SMR). When a person imagines movement, changes in SMR can be observed, typically in the alpha (α) and beta (β) frequency bands. These changes in SMR, within a specific frequency range, are referred to as Even-Related Desynchronization (ERD) and Event-Related Synchronization (ERS). ERD is a decrease in oscillatory activity, while ERS is an increase in oscillatory activity. Performing motor imagery can trigger changes in ERD and ERS[2].

The ERD/ERS patterns follow a homuncular organization[2], meaning that different parts of the motor cortex or different brain regions are associated with different parts of the body and the activities carried out by the limbs. Due to this fact, we can look at the changes in ERD/ERS for different movements. The activity of the hand MI is located on the contralateral side, meaning that MI of the left hand will produce most activity over electrode position C4, which is positioned on the right side. For the right hand, the change in activity will be most visible over electrode position C3, located on the left side of the motor cortex. Foot MI will mostly be invoked over the electrode position Cz, corresponding to the medial part of the motor cortex[2]. Even though the ERD/ERS pattern follows a homuncular organization we cannot separate every movement by analyzing EEG signals, this is due to the fact that some of the movements don't have large enough cortical areas. For instance, we are not able to separate the left foot from the right foot or separate different fingers. The left hand, right hand, foot, and tongue have large cortical areas and are also placed in different places on the cortex[2].

2.4 Preprocessing and Signal Decomposition

2.4.1 Filter

Notch Filter

EEG signals are affected by the non-physical artifact power-line noise which originates from the alternating main power supply. In Europe, this frequency is at $50Hz$, and can be removed by applying a notch filter on the signal. The notch filter works as a narrow band-stop filter, filtering out the given frequency.

High-, Low- and Band-pass Filter

The Mu- and beta-band is said to be two of the most important frequency bands in EEG signals[2], thus the need to filter a signal is prominent. To let through frequencies over a given threshold, a high-pass filter can be applied, while a low-pass filter can be applied to pass through frequencies below a given threshold. A band-pass filter will work as a combination of these and let through the frequencies between a given high and low threshold.

2.4.2 Rereferencing

When recording data by using EEG the purpose is to measure the electrical potential at the scalp where the electrodes are placed. One important point to take into consideration is that all measurements are relative, that is the voltage measured at a respective electrode is relative to a reference electrode. The reference electrode is supposed to be placed on a natural spot on the body, preferably on the head, which is not a simple task as the human body consists of much water and is very conductive. This means that the measurement will contain some noise. Rereferencing is a method used for changing the reference after the recording is finished in order to remove some of the noise introduced by the method of measurement[7].

When introducing rereferencing, a new reference is to be found after the end of the recording. This might be done by finding an average of several electrodes, or it can be composed of the recorded electrodes. The new reference found is subtracted from each of the electrodes. Thus, if a new electrode is chosen as the reference, its signal will become zero. After subtracting the new reference, the electrodes do now reflect the electrical potential between the new reference and the respective electrode[7].

Common Average Rereference

CAR is a method for increasing the Signal-to-Noise Ratio (SNR), by creating a global reference. The global reference is found by taking the sample-by-sample average of all the electrodes[8]. The calculation of the reference is given by the following equation [9]:

$$V_i^{CAR} = V_i^{ER} - \frac{1}{n}\sum_{j=i}^{n} V_j^{ER} \tag{2.1}$$

Where V_i^{ER} is the potential between the ith electrode and the reference and n is the number of electrodes used[9].

Laplacian Rereference

Instead of finding one common reference, the Laplacian rereferencing method finds a new reference for each electrode. The Laplacian rereferencing method works as a high pass filter, which enhances localized activities while suppressing the scattering activity[10]. The method uses an approximation of the second derivative. This approximation is done by first finding the weighted mean of the neighboring electrodes, where the weights are the distance between the electrodes, then subtracting the weighted mean from the electrode of interest[9]. The calculations follow the following equation from[9]:

$$V_i^{LAP} = V_i^{ER} - \sum_{j \in S_i} g_i j V_j^{ER} \tag{2.2}$$

where,

$$g_{ij} = \frac{\frac{1}{d_{ij}}}{\sum_{j \in S_i} \frac{1}{d_{ij}}} \tag{2.3}$$

S_i is the set of neighboring electrodes of the electrode i, for a small Laplacian placing scheme the set S_i is the nearest neighbors, and for large Laplacian, it is the set of next-nearest neighbors. d_{ij} is the distance between electrode i and j which will be the distance to either the nearest or the next-nearest neighbor, dependent on the size of the scheme[9].

2.4.3 Discrete Wavelet Transform

DWT is a linear method used for decomposing a non-stationary signal into different sub-bands of wavelets, where a wavelet is a waveform with an average value of zero and has a limited duration. In order to decompose the signal, it is necessary to choose a mother wavelet and the number of decomposition levels wanted[11]. Based on the decision of the mother wavelet, the trade-off between how compactly the basis function is localized in space and how their smoothness will vary. Based on the chosen mother wavelet, a family of wavelets will be made. This is done by scaling and dilating the mother wavelet[12]. This process will result in the signal being decomposed into a high-frequency part with detail coefficients and a low-frequency part with approximation coefficients. The low-frequency part of the signal is used for making the rest of the high-frequency levels until the predefined number of coefficients is reached[11]. In table 2.2 the resulting coefficients and frequencies of using the mother wavelet "bi-orthogonal 2.2" and four levels of decomposition of a signal with $512Hz$ sampling frequency are shown.

One of the advantages of using the DWT is that it captures both time and frequency. The wavelets are localized in space, which makes the functions and operators using wavelets become spars. This trait makes the wavelets well-suited for noise removal. Another advantage is that the use of wavelets gives a better resolution of different frequencies, due to the varying window size[12].

Table 2.2: Frequency sub-bands for four levels of DWT decomposition for sampling frequency at $512Hz$.

Four levels of decomposition	
Sub-band	**Freq. band** [Hz]
Detail coefficient, D1	$128Hz$-$256Hz$
Detail coefficient, D2	$64Hz$-$128Hz$
Detail coefficient, D3	$32Hz$-$64Hz$
Detail coefficient, D4	$16Hz$-$32Hz$
Approximation coefficient, A4	$0Hz$-$16Hz$

2.5 Feature extraction

2.5.1 DWT- and Bandpass-based Feature Extraction

Feature extraction can be done in several different ways. One method is to use the DWT decomposition method, where the signal is split into several frequency bands. For each of these frequency bands, a number of different features can be calculated[11]. These features will further be used in classification. Another method is to utilize a bandpass filter. By applying this bandpass-based feature extraction method, one single frequency band is extracted from the signals. On this band, features are extracted and further used in the classifiers[13].

When the signal is subjected to decomposition into subbands, either through the utilization of DWT or the bandpass method, a set of 16 features is computed on the resulting bands. These features are outlined below:

Instantaneous Energy

Instantaneous Energy (IE) is a feature that reflects the amplitude of the signal[14], and the energy distribution of each frequency band or decomposition level[11]. IE can be computed in the following manner:

$$f_j = log_{10}\left[\frac{1}{N_j}\sum_{r=1}^{N_j}(w_j(r))^2\right] \tag{2.4}$$

where $N_j = N/2^j$ is the number of samples for band j, and $w_j(r)$ is the wavelet coefficient at time r for band j[11].

Teager Energy

The Teager Energy (TE) feature does as IE, giving an amplitude analysis of the signal. Moreover, it reflects the variation in frequency[14]. TE can be computed in the following manner:

$$f_j = log_{10}\left[\frac{1}{N_j}\sum_{r=1}^{N_j-1}\left|(w_j(r))^2 - w_j(r-1)\cdot w_j(r+1)\right|\right] \tag{2.5}$$

where $N_j = N/2^j$ and $w_j(r)$ reflects the same as for IE[11].

Mean

Mean is the average value of all the amplitude values across a specific time window of a given electrode and is expressed as:

$$\mu = \frac{\sum_{i=1}^{N}\text{amplitude}_i}{N} \tag{2.6}$$

with N as the number of samples in the time window.

Variance

The variance of a signal is how much each point in a signal differs from the mean, given as:

$$\text{Var}(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2 \tag{2.7}$$

where N is the total number of data points in the signal, x_i is the i-th data point, and μ is the mean.

Standard Deviation

Standard Deviation (SD) is the variability or spread in the data following the normal distribution. A lower SD indicates that the data points are less spread out and that the data points are more tightly clustered around them. SD can be represented as:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(\mu - x_i)^2}{N}} \tag{2.8}$$

where N is the total number of data points, x_i is the i-th data point, and μ is the sample mean[15].

Root Mean Square

Root Mean Square (RMS) is a way of calculating the average of values of a signal at a given electrode, by taking wach amplitude value squared and averaging over a period of time. The equation is as follows:

$$\text{RMS} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i)^2} \tag{2.9}$$

where N is the total number of data points in the signal and x_i represents the value of the i-th data point.

Peek-To-Peek

Peak-To-Peak (PTP) is the difference between the highest and the lowest value of a signal and can be expressed as

$$\text{PTP} = \max(x) - \min(x) \tag{2.10}$$

where x represents the signal.

Skewness

Skewness is a statistical measure of the asymmetry of the signal distribution, given as:

$$\text{Skewness} = \frac{1}{N}\sum_{i=1}^{N}(\frac{(D_i - \mu}{\sigma})^4 - 3 \tag{2.11}$$

with D_i describing each band, N as data points, μ is the mean and σ the standard deviation[16].

Kurtosis

Kurtosis is also a statistical-based feature. The kurtosis describes the shape of a probability distribution. For EEG signals, the kurtosis describes the peakedness or flatness of the distribution of the amplitudes[17].

$$K = \frac{E[(x-m)^4]}{E[(x-m)^2]^2} \tag{2.12}$$

Where E is the expectation value, and m is the mean of the signal.

Hjorh Mobility

Hjort Mobility (HMO) is a way to indicate the statistical property of EEG signals in time domain and represent the proportion of standard deviation of power spectrum. The equation is as follows[18]:

$$HMO = \sqrt{\frac{var(y'(t))}{var(y(t))}} \tag{2.13}$$

Hjorth Complexity

Hjort Complexity (HCO) is a measure of change in frequency that indicated the resemblance of a signal's shape to that of a pure sine wave. When the signal closely resembles a sinusoid, the HCO value converges to 1. The equation for HCO is as follows[18]:

$$HCO = \frac{HMO(y'(t))}{HMO(y(t))} \tag{2.14}$$

Higuchi Fractal Dimension

The fractal dimension can be used to describe the complexity of a signal. The Higuchi Fractal Dimension (HFD) can be used to characterize non-linear and non-stationary data[14]. It estimates the dimension of a time-varying signal directly in the time domain. The algorithm does this by approximating the mean length of the curve using segments of k samples[14]. HFD can be calculated in the following manner:

$$H = \frac{ln(|L(k)|)}{ln(\frac{1}{k})} \tag{2.15}$$

where $L(k)$ is an array with the mean length of curve k defined as the following:

$$L(k) = \frac{1}{k}\sum_{m-1}^{k} L_m(k) \tag{2.16}$$

where the $L_m(k)$ is the length of the curve, and the mean is calculated for the curve for each time interval k [14].

Petrosian Fractal Dimension

Petrosian Fractal Dimension (PFD) is a rapid method for calculating the Fractal dimension and is done by translating the series into a binary sequence. Then the fractal dimension is computed as follows:

$$P = \frac{\log_{10}(n)}{\log_{10}(n) + \log_{10}(\frac{n}{n+0.4N_\Delta})} \tag{2.17}$$

where n is the length of the sequence and N_Δ is the number of sign changes in the binary sequence.

Katz Fractal Dimension

Katz Fractal Dimension (KFD) measures the complexity or irregularity of a signal. It compares the actual number of smaller units that make up the curve with the minimum number of units needed to replicate a pattern of the same size. KFD can be expressed as:

$$KFD = \frac{\log_{10}(L)}{\log_{10}(d)} \tag{2.18}$$

where N is the length of the curve or sum of distances between successive points, and d is the diameter, which is calculated as the estimated distance between the first point of the sequence and the point of the sequence that provides the farthest distance[19].

Sevcik Fractal Dimension

Sevick Fractal Dimension (SFD) is a way of calculating the fractal dimension of a signal. N values are sampled from a signal and subjected to a double linear transformation. This transformation maps the signal into a unit square. Then, the total length (L) of the curve is found by taking the normalized abscissa of the square, which is used with the total number of values N to calculate the SFD value. This calculation is as follows[20]:

$$SFD = 1 + \frac{ln(L) + ln(2)}{ln(2 * N)} \tag{2.19}$$

Line Length

Line Length (LL) is a geometry-based feature that measures the complexity of the signal. A high value indicates a more complex and varying signal. It is calculated by taking the absolute value of all distances between successive points:

$$L = \frac{1}{N} \sum_{k=1}^{N-1} |x(k-1) - x(k)| \tag{2.20}$$

where x is the EEG signal, k represents a discrete number indexing the time, and N is the total number of points in the EEG window[21].

2.5.2 Common Spatial Pattern

The CSP algorithm is used, with high success, for calculating spatial filters for detecting ERD and ERS. By implementing CSP the objective is to find features that can discriminate between different classes of EEG signals. The CSP algorithm gives the filtered signal Z which maximizes the difference in the variance of the different classes of the EEG measurement[22]. The signal Z is given by the following equation,

$$Z = W^T E. \tag{2.21}$$

E is the raw EEG signal and W is the projection matrix. The columns in the W are the spatial filters, and the columns in W^{-1} are the spatial patterns[23]. To maximize the difference between the variance of the classes we calculate the covariance for the different classes and then combine them, this is done in the following manner[23],

$$R_{c1,c2} = \frac{1}{N}\sum_{n-1}^{N}\frac{E^n(E^n)^T}{Tr(E^n(E^n)^T)} \tag{2.22}$$

$$R_c = R_{c1} + R_{c2}. \tag{2.23}$$

For the following step, a whitening matrix, P, from the eigenvalues of the covariance matrix is constructed,

$$P = \sqrt{\Lambda_c^{-1}}U_c^T, \tag{2.24}$$

where U_c is the matrix containing the eigenvectors and Λ_c is the matrix containing eigenvalues. With the whitened covariance matrices and the P matrix, we get the following,

$$S = PRP^T \tag{2.25}$$

The further eigen-decomposition of S can be expressed as,

$$S = B\Phi B^T, \tag{2.26}$$

with Φ as the eigenvalue matrix, and B as an optimization variable for maximizing the difference between the classes by finding the spatial filters, V, where each of the columns is a filter. From V, filters are selected to obtain the spatial filter W. The matrix V can be found in the following matter,

$$V = B^T P. \tag{2.27}$$

The CSP algorithm was originally only made for binary classification, but later extended to be used in classification for multiple classes[23].

2.6 Classification

2.6.1 Random Forest

RF is a tree-based ensemble classifier. Each tree in the classifier is dependent on a collection of random variables. The classifier takes an input vector that represents the real-valued data with a corresponding vector containing the real-value response (classes) to the input vector. The RF classifier aims to find a prediction function for classifying the input, this prediction function is found by minimizing a loss function[24].

The RF classifier consists of a forest of trees, where each tree in the forest is based on binary recursive partitioning trees. Each tree starts with a root, which is the first node of the tree. This node is split into two descendant nodes, according to the value of one of the predictor variables. This is done for every descendant node until it reaches the terminal nodes[24].

The RF algorithm uses "Random Sampling", meaning that the algorithm randomly selects a subset of the training data with replacement. This is also known as bootstrapping. Bootstrapping causes every tree in the forest to be trained on a different subset of the data, making the model less sensitive to the training data. For each tree, a random subset of the features is selected to split the node.

To classify new data points, the data is passed to all the decision trees in the forest. Further, all votes for each class are summed up, where all the class predictions from each tree are independent of each other. By using the majority rule, the class with the highest number of votes is assigned to the data point as its class. With the independence of the trees and the use of the majority rule, the model is less prone to errors and overfitting.

2.6.2 Gradient Boosting

The GB algorithm is like the RF a tree-based ensemble classifier. The basic idea behind the GB classifier is to sequentially add new models to the ensemble, where each model is a decision tree that tries to correct the mistakes made by the previous models (i.e. it combines multiple weak classifiers to create a strong one). This way, the final model is a weighted combination of all the models, with each model contributing a small amount to the final result[25].

Initially, the algorithm starts with one initial decision tree that predicts the average value of the target. After the initialization, the algorithm trains a sequence of models/trees iteratively. Each tree is built based on the residuals from the previous model, using the different features at the nodes, and separating the observations into branches. The residuals are calculated by exploiting the gradient of the loss function used, and for each iteration, the models try to correct the error from the previous iteration.[25]

After all the models are trained, a combination of the models is made for making a final classification. This model is found by calculating a weighted sum of all the models in the ensembles, with the weights determined by the performance of the model. To classify a new data point, the GB classifier does as the RF classifier and runs the data point down every tree in the ensemble. Then, for making the classification, it uses the weighted sum of all the predictions obtained from the ensemble.[25]

2.6.3 Linear Discriminant Analysis

The LDA classifier tries to find a subspace of lower dimension, compared to the original data sampled dimension. The original data points need to be separable, which is measured by the use of mean and variance [26]. In classification, LDA tries to find a linear combination of features that maximizes the separation between different classes of data. Originally the classifier was made for binary classification and is done by the following matter [26]:

First, for each class, A and B, the sample means can be found:

$$\bar{x}_{A,B} = \frac{1}{N_{A,B}} \sum_{x \in A,B} x \tag{2.28}$$

where N_A and N_B is the number of samples in class A and B.
Next, to express the sample variability of each class the positive definite scatter matrices are found:

$$S_{A,B} = \sum_{x \in A,B} (x - \bar{x}_{A,B})(x - \bar{x}_{A,B})^T \tag{2.29}$$

The hyperplane, defined by vector ϕ, is found such that when the data samples are projected the variance will be minimal:

$$\min_{\phi} = (\phi^T S_A \phi + \phi^T S_B \phi) = \min_{\phi} \phi^T (S_A + S_B)\phi = \min_{\phi} \phi^T S \phi \tag{2.30}$$

The scatter matrix between the two classes can be found mathematically as:

$$S_{AB} = (\bar{x}_A - \bar{x}_B)(\bar{x}_A - \bar{x}_B)^T \tag{2.31}$$

Now, according to Fisher's intuition, the wish is to find a hyperplane maximizing the distance between the means between the two classes and at the same time minimizing the variance in each class. Fisher's criterion can be formulated as follows:

$$\max_{\phi} \frac{\phi^T S_{AB} \phi}{\phi^T S \phi} \tag{2.32}$$

Replacing the denominator with an equality constraint, $\phi^T S \phi = 1$, this becomes a convex optimization problem with a global optimum.
To further expand this classification method into a multiclass problem, we need to redefine the scatter matrix, $S = S_1 + \cdots + S_n$, where the inter-class matrices are defined as follows:

$$S_{1,\cdots,n} = \sum_{i=1}^{n} p_i (\bar{x}_i - \bar{x})(\bar{x}_i - \bar{x})^T \tag{2.33}$$

with p_i as the number of samples in *ith* class, $\bar{x}_i$ as the mean for each class and the $\bar{x}$ as the total mean calculated as:

$$\bar{x} = \frac{1}{p} \sum_{i=1}^{n} p_i \bar{x}_i \tag{2.34}$$

Lastly, from this we can obtain ϕ by solving the following eigenvalue problem:

$$S_{1,\cdots,n}\phi = \lambda S\phi \tag{2.35}$$

2.6.4 Support Vector Machine

SVM tries to find a hyperplane (i.e., a decision boundary) that maximally separates the data points into different classes. The SVM classifier separates the classes by using a kernel to transform the input data into a higher dimensional space, where the different classes are linearly separable. After transforming the data, the classifier tries to find a hyperplane that maximizes the distance between the hyperplane and the closest data points. The distance is called the margin, and the data points closest to the hyperplane are called support vectors[27].

When the data points are not separable by a hyperplane, a "soft margin" approach is introduced. This "soft margin" approach includes a penalty term to the SVM classifier and makes the classifier allow some misclassification. For the SVM classifier to be able to handle both linear and non-linear classification problems, different kernels can be used for different problems, where the kernels add a dimension to the data. The kernels do also add the benefits of being able to use input data, not in vector format, and also being able to combine different types of data [27].

Overall, the SVM classifier tries to find the hyperplane that not only separates the data points but also generalizes well to new data. This is achieved by minimizing the classification error and maximizing the margin simultaneously. The SVM is a powerful classification algorithm that can handle complex datasets and generalize well to new data[27]. Figure 2.5 shows an illustration of the SVM classifier.

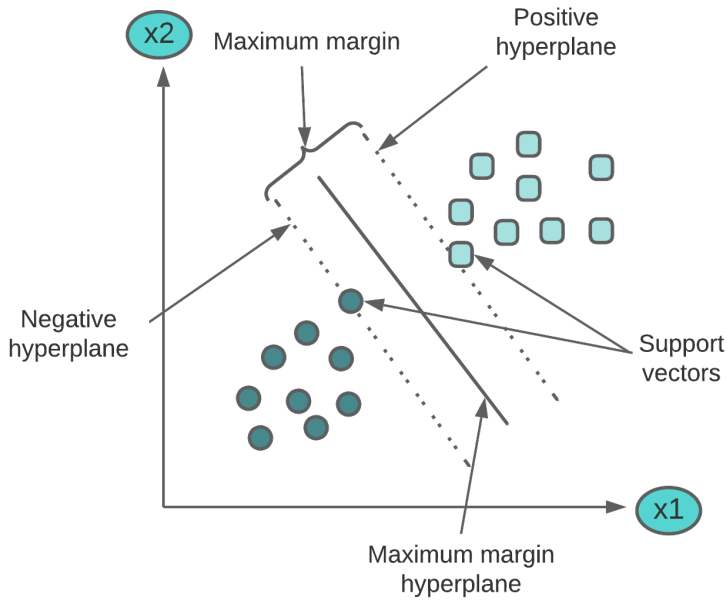


Figure 2.5: Simple illustration of how the SVM classifier works, with respective support vectors, hyperplanes, and margins.

2.6.5 Convolutional Neural Network

Over the past few decades, DL and Deep Neural Network (DNN) have grown to be one of the most powerful tools in classification[28]. DNNs are collections of neurons organized in a sequence of multiple layers. The input to the neurons in a layer is the neuron activation from the previous layer, making a complex nonlinear mapping from input to output. The weights of these neurons are adapted when feeding it with the data by backpropagation[29]. One of the most commonly used DNN is the Convolutional Neural Network (CNN), which have shown great results in various application including image detection, time series data, and video detection[28].

A CNN can process data with a known, grid-like topology. It employs a mathematical operation called convolution, which is a linear operation between the input and a kernel. The output of this convolution is a feature map containing a map of activations indicating the input feature's location and strength[30]. The convolution is done to decrease the complexity and number of connections in the network[31].

In contrast to traditional ML, CNN does not require the signal to be preprocessed in any way. The CNN classifier can use the raw, denoised, or filtered signal to extract features and make a classification based on the weight and biases learned during training[32]. Even without feeding the CNN with features, it has shown to outperform traditional ML in many applications[32][33], including EEG classification[31].

EEGnet by Lawhern et al.

EEGnet is a compact CNN specifically designed for the classification of EEG signals. The network contains input and output layers in addition to two hidden layers named blocks. The blocks perform depthwise and separable convolution to construct a model which encapsulates well-known EEG feature extraction concepts for BCI[34]. The Depthwise convolution convolves each input channel with a separate filter, in contrast to normal 2D convolution where the convolutions are performed over all or multiple input channels[35]. The separable convolution is a depthwise convolution followed by a pointwise convolution[34], where a pointwise convolution applies a single filter to each element of the input feature map[36].

In addition to having convolutional steps in the blocks, they also include batch normalization and average pooling[34]. Batch normalization normalizes the input to each layer, making the training process more stable and efficient, while avoiding the output of each layer to become too large or small. Average pooling is a method that downsamples the feature map by computing the average of values within a window and using the resulting values in the output map. Thus, the feature map has a smaller spatial dimension, improving computational cost and reducing the chance of overfitting[30]. To complete the network, dropout is included to avoid overfitting[34]. Dropout is a regularization method that randomly deactivates a percentage of the nodes during each training step[30].

The EEGnet is said to be robust as it is able to learn a wide variety of interpretable features from many different BCI tasks. It has also been shown that it can outperform other well-known CNN classifiers for EEG classification since it performs well on both small and large datasets[34].

2.6.6 Transfer Learning

ML and DL are widely used and successful methods in many domains and applications where past information (training data) can be used to predict an outcome[37]. These methods have traditionally been designed to work in isolation and to solve a specific task. Thus, they have to be rebuilt every time one of the feature-space distributions changes. To overcome these problems, TL utilizes knowledge acquired for one task to solve related ones[38]. Thus, TL can be described as a situation where what has been learned in one setting, can be used to improve generalization in another setting[30].

The need for TL occurs when there is a limited supply of target training data. Training on a small amount of data can lead to a bad classification, thus transferring knowledge from one model to the other can enhance accuracy[37]. As well as being convenient when having a small amount of data, TL can increase computational efficiency by either saving time and resources from having to train multiple ML models from scratch or reducing the effort of expensive data collection and labeling[39].

TL can be divided into three different learning strategies, based on if there is labeled data or not[39]. This can be seen in figure 2.6, where source domain refers to the data used to train the model before the TL, and target domain refers to the data we want to classify by using TL. In inductive TL, the target task is different from the source task, requiring labeled data from the target domain to induce the model for use on the target domain. On the other hand, in transductive TL the source and target tasks are the same while the domains are different. Unsupervised TL has the same source and target domains and different tasks, but the labeled data is unavailable in either of the domains[39].

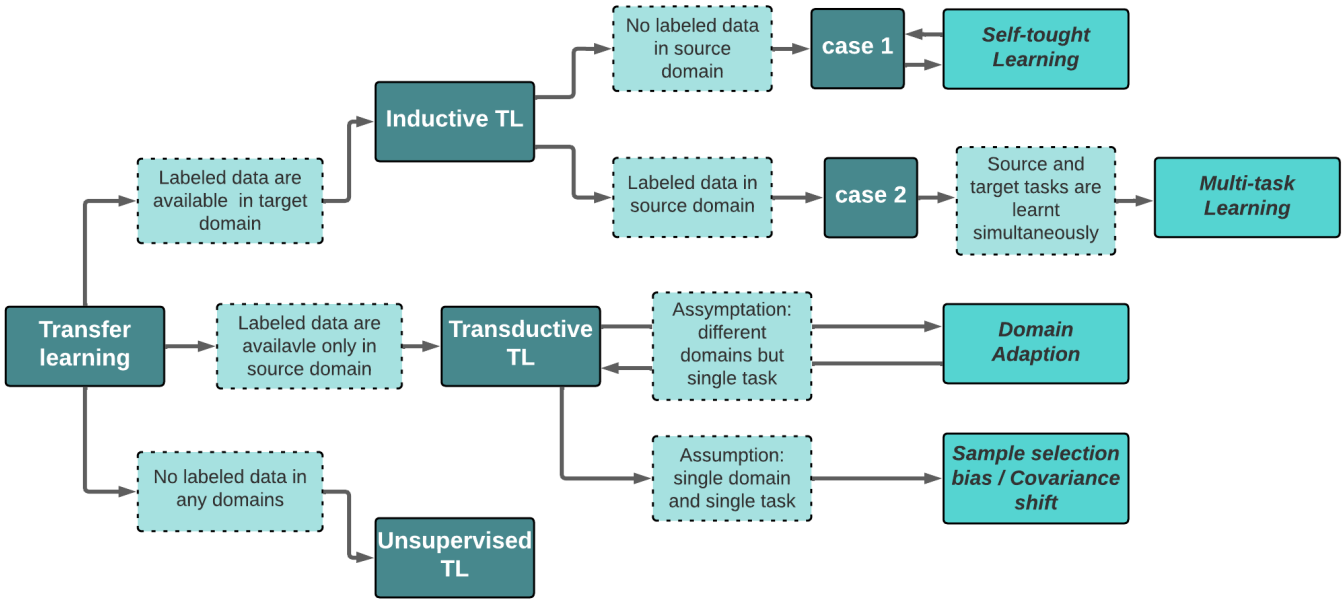


Figure 2.6: Learning strategies for transfer learning. Adapted from [39].

2.7 Non-dominated Sorting Genetic Algorithm

The NSGA is a type of Genetic Algorithm (GA). The GAs are inspired by Darwin's theory about natural evolution and natural selection also called "Survival of the fittest" [40]. Natural selection is based on the idea that the "fittest" individuals from the populations will produce offsprings that will inherit the characteristics from the parents. The next generation will evolve further, based on the genes of the previous generation. This evolution and replacement of the generations are done iteratively by using the genetic operators, selection, mutation, and crossover [40]. The population of a GA consists of different chromosomes, where the chromosomes consist of 0 and 1. Each of the chromosomes gets a value from the fitness function, and the value assigned decides which of the chromosomes will be selected for the development of the next generation[40].

Non-dominated sorting is based on the Pareto dominance principle and is used for sorting the solutions in populations [41]. This is an important process in the selection operation of many multi-object GAs[41]. The Pareto optimal solutions are solutions that are superior to those in the search space when all objectives are taken into consideration, but inferior to the other solutions for one or more objectives[11]. These solutions are also called non-dominated solutions, and the rest of the solutions are called dominated solutions[11].

The procedure of the NSGA-III is described in [42] and shown in figure 2.7 and can be described in the following manner: First, the algorithm generates a random initial population of parents, P_t, and sets a uniform reference point. The population of the next generation, Q_t is then found by combinations and mutations of different parents. After the new generation is found, this population and the parent population are combined into a new population, R_t. Then non-dominated sorting is applied to divide the population into non-dominated sorting levels, and from these levels selects individuals to make a new population S_t. The new population S_t is the new parent population for the next iteration of the algorithm. For choosing the K individuals shown in figure 2.7, some additional steps are done. Firstly normalization of the objective values of the individuals in S_t, then it defines reference lines. After defining the reference line, it calculates the perpendicular distances between the individuals in S_t and the reference lines. Each individual in S_t is associated with a reference point according to the minimum perpendicular distance. Then it chooses the reminding individuals based on the calculated niche count.

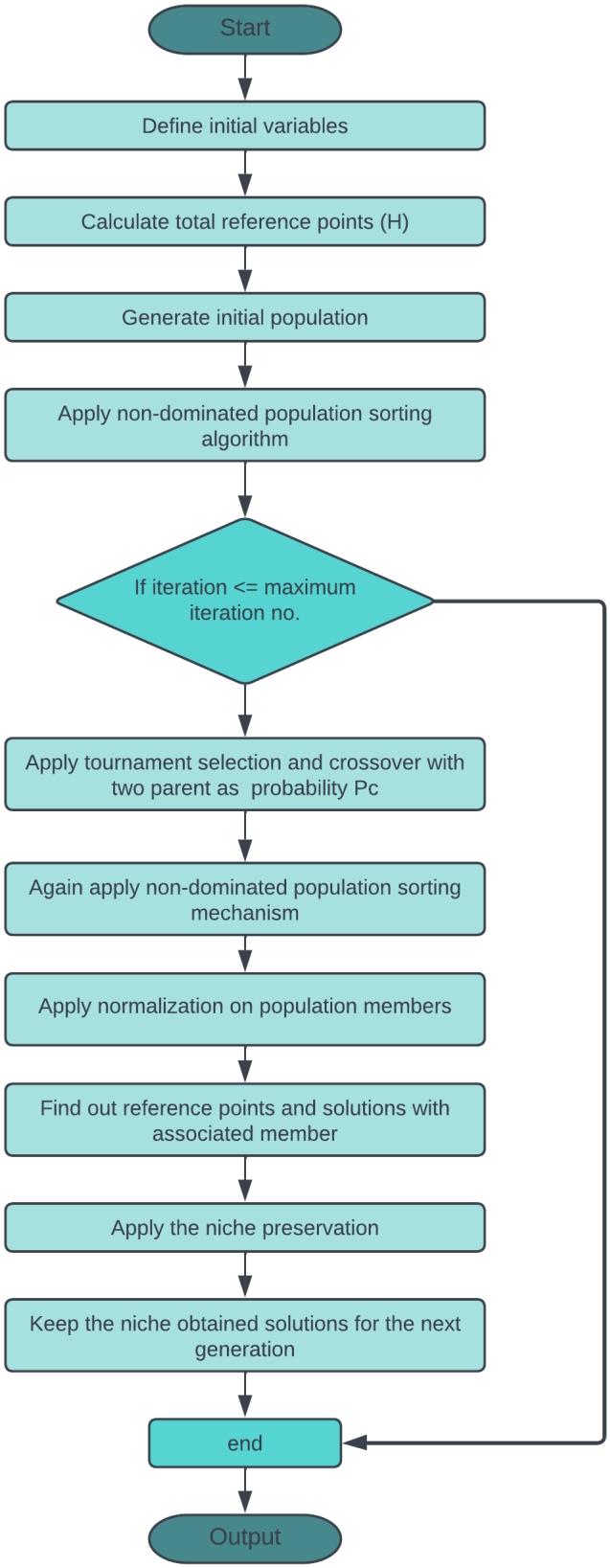


Figure 2.7: Flow chart of the NSGA algorithm adapted from [42].

Chapter 3

Literature Review

MI-classification is a field in EEG-studies that is well-researched. To get an overview of this field, different meta-studies are reviewed. These meta-studies offer valuable insight into which pipelines have been explored and identify those that have exhibited favorable outcomes in terms of MI classification using ML and DL techniques. In addition, a review of the dataset is given to give a presentation of the results obtained from the different subjects in the dataset.

3.1 State-of-the-art in Motor Imagery Classification

3.1.1 Preprocessing

In order to reduce issues related to memory requirements, computational time, system complexity, and overfitting, fewer EEG electrodes are preferred[43]. While most studies on MI classification employ all available electrodes within the dataset, research in this area suggests that conducting an electrode selection might either enhance the performance or not affect it in any significant way[44][45]. In fact, findings from a survey study[46] on electrode selection conclude that the number of electrodes can on average be reduced by up to 80% without significantly affecting classification tasks.

In addition to electrode selection, features must be extracted from the signal before classification. According to Aggarwal and Chugh[47], CSP and its variants are widely used in MI-based BCIs. Furthermore, studies have shown that filter bank Common Spatial Pattern (fbCSP) consistently yields the best performance in MI classification when compared to other methods relying on manual feature extraction[43].

Time-frequency domain feature extraction methods are also widely employed in MI-BCIs, as they enable spectral information about the EEG signal. DWT is one of these methods which proves to be a powerful tool due to its ability to capture different information about MI actions within the subbands[48]. Frequently, band power features are calculated from these bands, as they effectively capture the change in EEG rhythm amplitudes[49].

The majority of the MI-signals typically lay in within the frequency range of $8-30Hz$[2]. For this reason, multiple studies have shown that using a filter to extract these frequencies can enhance the results of MI classification[13][43]. In addition, frequency filtering can serve as an effective way of removing a large portion of the noise. Specifically, it can remove low-feature artifacts such as eye blinking, and high-feature artifacts such as electromyography[43].

3.1.2 Classification

Machine Learning

Numerous ML algorithms have been employed in the classification of EEG-BCI signals, some with greater success than others. SVM and LDA are two of the classifiers that have shown great results, where SVM often outperforms most classification methods[49][47]. SVM has multiple times shown better results than LDA when classifying EEG signals, but newer studies show that adding a regularisation on LDA will improve the classification, making it just as good or better than regular LDA and SVM[47][50][51].

In recent years, there has been a growing interest in tree-based classification methods, as they have shown to outperform other popular classification algorithms, specifically in scenarios with limited available data[49]. In particular, multiple studies have shown that RF can outperform both SVM and LDA when it comes to MI classification[52][53][54]. In addition, Xu et al.[55] showed that using GB increased the accuracy by 13% compared to Long Short-Term Memory (LSTM) Recurrent Neural Network (RNN) for MI tasks, while Mirzaei and Ghasemi[56] showed that GB can outperform both SVM and LDA.

Another method that has shown great results in multi-class MI classification is to use multiple classifiers rather than one. Studies have shown that a k-Nearest Neighborhood ensemble method can significantly improve the accuracy compared to conventional models[47]. The method of using an ensemble of classifiers has also shown superior results in online classification, according to Lotte et al.[49]. Lotte et al. also state that adaptive LDA has been explored successfully in online classification.

Deep Learning

There are multiple DL algorithms tested on EEG signals. However, according to Altaheri et al.[43], the most commonly adopted method is CNN based. Altaheri et al.[43] further state that CNN often outperforms other DL techniques when classifying MI signals[43]. Wang et al.[57] demonstrate that using a CNN outperforms LSTM by an average of 6.55% to 12.54%. Furthermore, another study shows that CNN outperforms both LSTM and RNN, stating that one of the reasons CNN outperforms LSTM may lie in the possibility that LSTM can be prone to overfitting[58].

In addition, Wang et al.[57] show that CNN outperforms SVM by an average of 3.07% to 9.05%. Another study concludes that CNN can outperform regular ML techniques that employ fbCSP as feature extraction and LDA or SVM as classification techniques[59]. Several studies conclude the same, that CNN often achieves better performance than standard ML[32][60].

Subjects performing MI tasks can be divided into high and low performers based on the ease of building a classifier and the results obtained. According to Tibrewal et al.[32] using a CNN can improve classification significantly more for low performers than for high performers, showing that Neural Network (NN) is an important tool for MI classification. Furthermore, Lotte et al.[49] state that DL and NN can compensate for different struggles encountered from EEG signals. Specifically, DL can improve classification accuracy when the signal-to-noise ratio is low and the BCIs have poor reliability.

Transfer Learning

TL is a valuable approach when there is not enough labeled data to train a model, but there exists a model where the knowledge of a domain can be transferred to the new domain. Wan et al.[61] describe the advantages and challenges of four main methods of TL and explores their practical application in EEG signal analysis in recent years. The four methods are domain adaption, improved CSP algorithms, DNN methods, and subspace learning. Their research found that the main advantages of TL in EEG signal analysis are that it reduces the requirement for data and that TL can make the model more flexible, such that it can match different individuals and tasks through adjustments[61]. On the other hand, both Wan et al.[61] and Lotte et al.[49] state that negative transfer is a challenge that must be faced, and that TL should be used with care, as it also may decrease performance.

To minimize the risk of negative transfer, Chen and Lu[62] purpose a method where they select appropriate sources to borrow knowledge from when doing the transfer. Their results show that the classification accuracy improves by 12.72% compared with the non-transfer method. Another study done by Ling and Jung[63] purpose a conditional TL to facilitate a positive transfer for each subject. Their method uses data from other subjects with similar EEG signatures to transfer knowledge. The results show that multiple subjects could benefit from TL, with some subject accuracies increasing from chance level (50%) up to 70%. In addition to these studies, multiple papers conclude that using TL on EEG data can increase the accuracies compared to the subject-dependent counterpart[59][64][65].

3.2 Review of Results from Dataset

A dataset collected at the Laboratory of Brain-Computer Interfaces at Graz University of Technology in Austria is used in this thesis. It was collected for the international BCI conference in 2014 by David Steyrl, Reinhold Scherer, Oswin Förstner, and Gernot R. Müller-Putz[66].

Steryl et al.[67] made a paper on the subjects' performances in this dataset. The pipelines used to find the performances are fbCSP combined with RF and fbCSP combined with Shrinkage Linear Discriminant Analysis (sLDA) for offline classification, and Discrete Fourier Transform (DFT) combined with RF for online classification. Of the three pipelines, offline classification using fbCSP and RF gave the best results, with median accuracies ranging from 53.33% on subject 14 to 100.00% on subject 1. On offline classification RF gives on average 2.50% higher peak accuracy and 2.90% higher median accuracy than using sLDA. When using online simulation, the peak accuracies are above chance level for all subjects, ranging from 65.00% to 100.00%, showing that RF can be used for online classification[67].

From the paper of Steryl et al.[67], it can be seen that subjects S11-S14 have some of the lowest results, with median accuracies ranging from 52.50% to 88.33% using the best pipeline. These subjects are the focus of a paper written by Wang, Yang, and Huang[68]. Their results range from 67.00% to 87.00%, giving higher average accuracy than Steryl et all. There are three pipelines tested in this paper, where Eucledian Space Data Alignment is used as preprocessing on all of them. The tree pipelines consist of fine-tuned CNN, CSP-SVM, and CSP-2DCNN, where CSP-2DCNN give the highest results. The performance of this method is between 7.00% to 15.00% higher compared to CSP-SVM and 4.00% to 13.00% higher than fine-tuned CNN[68].

A third paper uses two different feed-forward NN, namely Multi-layer perceptron (MLP) and Extreme Learning Machine (ELM), to obtain results. Both of the pipelines use a method based on Riemannian geometry obtained from 15 frequency bands from $8-24Hz$ to calculate features that are fed to the NNs. The results show that MLP has a higher kappa value with an average lower SD than ELM, making it better at classification. They also show that there is a statistical difference across the subjects and that the number of hidden units in the classifiers will affect the results[69].

Chapter 4

Data Acquisition and System Design

4.1 Equipment

The drone used in the experiments is a Crazyflie 2.1 designed by Bitcreaze. A picture of the drone can be seen in figure 4.1. To enable control over the drone, a Python-based Application Programming Interface (API) is developed, enabling the possibility to transmit commands to the drone in real-time. Furthermore, a communication interface named Crazyradio PA serves as the conduit for establishing a wireless link between the drone and the computer system.

Figure 4.1: Picture of the drone Crazyflie 2.1.

4.2 Electrode Placement

Steyrl et al.[67] employs the widely adopted 10-20 system for electrode placement in their study. Specifically, the three central electrodes, named C3, Cz, and C4, are positioned according to this system. In conjunction with these center electrodes, four additional electrodes are positioned at a distance of 2.5 cm from each center electrode, as seen in figure 4.2. These supplementary electrodes can be used as a derivation of a Laplacian rereferenicing configuration for the central electrodes, or as extra electrodes when doing classifications. When the Laplacian rereferencing scheme is used, only the center electrodes are used for classification.

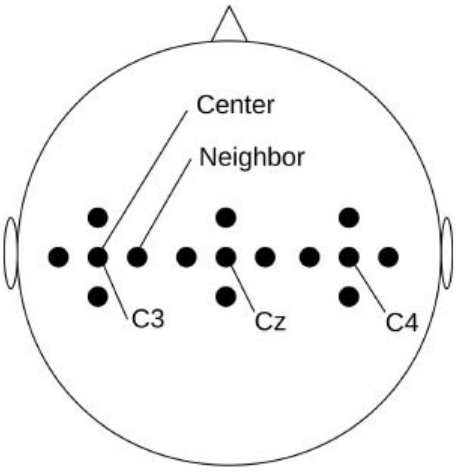


Figure 4.2: Small Laplacian electrode placing scheme, centered at electrodes C3, Cz, and C4.Reprinted from [67].

4.3 Data Epoch

4.3.1 Offline Epoching

The dataset used in this thesis[66] consists of eight runs per subject. Each run consists of 20 trials, with each trial spanning a duration of approximately 10 to 11 seconds. Specifically, in each trial, a period of 5 seconds is marked as MI-task, where the first second is when the cue is shown. One second before the cue, there is a beep to alert the subject of a new task. The remaining duration of the trial is marked as rest. For the offline experiments, a duration of 3 seconds of task and rest data is used for analysis. This can be seen in figure 4.3.

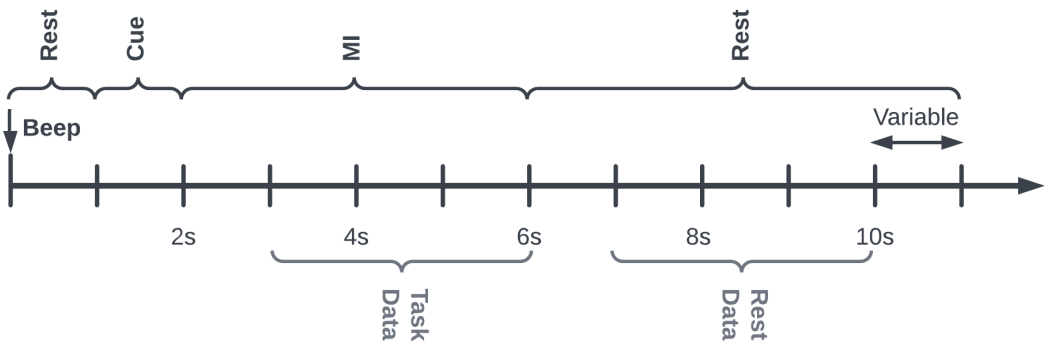


Figure 4.3: Visualization of offline epoching of data. Adapted from[67].

4.3.2 Online Epoching

The EEG recordings produce a continuous stream of data, so segmentation is needed in order to align the data stream with the structured format of the training data used to make the classification model. For the online classification, a time window optimization is employed to determine the optimal second within the training data that yields the highest accuracy. Further, the stream can be continuously segmented into data of 1 second corresponding to 512 data points. This will lead to new classifications roughly every second, as illustrated in figure 4.4

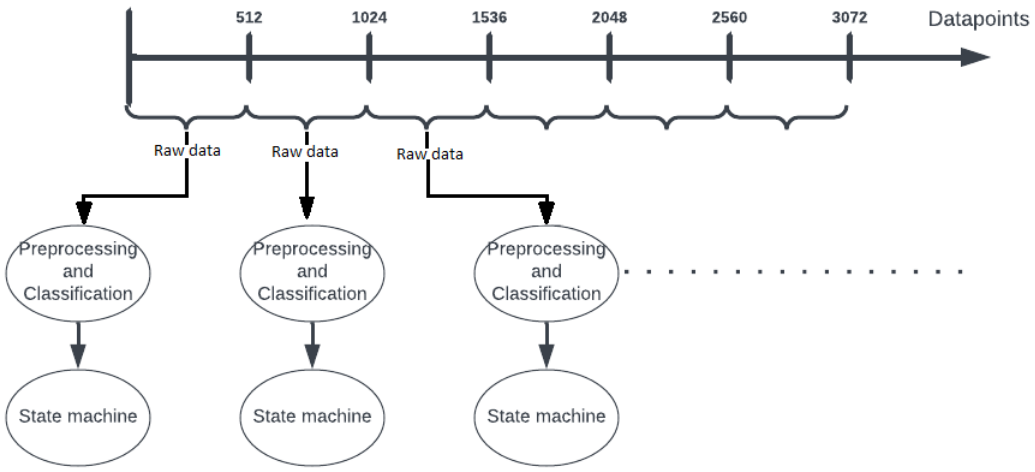


Figure 4.4: Visualization of online epoching of data.

4.4 Data Flow

4.4.1 General Data Flow

The flow from raw EEG data to classification is shown in figure 4.5. As seen from the figure, the initial step involves preprocessing of the raw EEG data, before the pipeline diverges into two branches based on the employment of either DL or ML approaches. It is necessary to diverge the pipeline due to ML algorithms' need for feature extraction before classification, which is not needed before the DL classification. Further, in the ML branch, the data undergoes either DWT decomposition or bandpass filtering, followed by feature extraction and classification. The DL branch undergoes an optional signal filtering, after which the data is fed directly into the CNN classifier for classification.

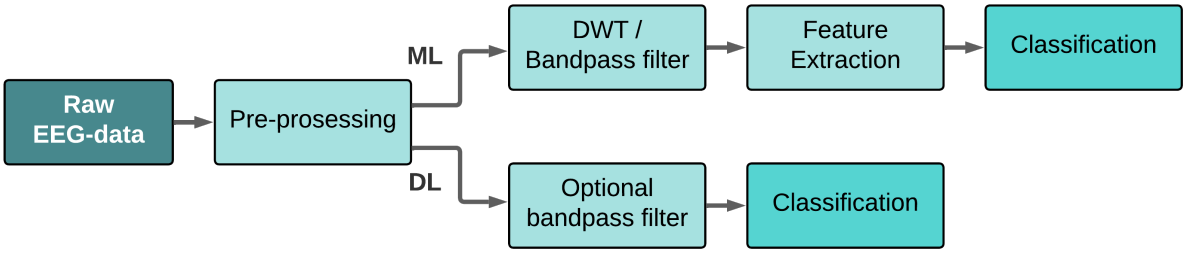


Figure 4.5: Flowcharts describing the steps made from EEG signal to classification.

4.4.2 Machine Learning Flow

To prepare the signal for classification, a preprocessing technique is employed. This phase encompasses a notch filtered at $50Hz$ to remove the electrical interference and an optional rereferencing of the signal. Two different rereferencing methods are tested: CAR and Laplacian rereferencing. Following this stage, the signal undergoes either decomposition or bandpass filtering. For the DWT decomposition, 4 levels of decomposition and the mother wavelet "Biorthogonal 2.2" is used. This results in the sub-bands shown in table 4.1. When the signal is bandpass filtered, the frequency range is typically set to $8-30HZ$, unless anything else is given.

Table 4.1: Frequency sub-bands for four levels of DWT decomposition for sampling frequency at $512Hz$

Four levels of decomposition	
Sub-band	**Freq. band** $[Hz]$
D1	$128Hz$-$256Hz$
D2	$64Hz$-$128Hz$
D3	$32Hz$-$64Hz$
D4	$16Hz$-$32Hz$
A4	$0Hz$-$16Hz$

Two methods are used for feature extraction, CSP and manual calculation of 16 different features. The application of CSP is exclusively employed following bandpass filtering, whereas the manual feature calculation is conducted on both the bandpass-filtered and DWT-decomposed signals. In the case of DWT, the 16 features are computed for each decomposition level, whereas for the bandpass filtered signal, they are computed for the entire frequency band. From these 16 features, the NSGA algorithm is used to find a subset of features that lead to enhanced accuracy in general across the subjects.

The classification stage involves the evaluation of four distinct classifiers, namely RF, GB, SVM, and LDA, as they have shown promising results on MI classification. While multiple parameters can be chosen for each classifier, the default values are primarily utilized in this study. In the SVM classifier, a linear kernel is employed to facilitate binary classification. This way the data can be split into task and rest, and further into the two different MI-tasks.

To minimize the potential of any dependency in the classifiers or dataset, the data is shuffled, and 10-fold cross-validation is performed. This approach involves dividing the dataset into ten equal-sized subsets, iteratively training the classifier on nine subsets while validating the performance on the remaining subset, and repeating this process ten times with different partitioning combinations.

4.4.3 Deep Learning Flow

The same preprocessing methods employed for the ML classification are also applied to the DL classification. Following preprocessing, the signal undergoes bandpass filtering, within the frequency range of 8-30Hz, unless otherwise specified, before being inputted into the CNN classifier known as EEGnet. Even though the EEGnet has several parameters, the default values are used in the experiments. Moreover, for the compilation of a NN in general, a loss function and optimizer must be chosen. The *Adam* optimizer is employed, as it has shown high effectiveness across various applications. The choice of loss function depends on the number of classes present in the data. For binary classification scenarios, the *Binary Cross-Entropy* loss function is employed, whereas for multi-class classification, the *Categorical cross entropy* loss function is utilized.

During the training of the CNN, a batch size of 16 is used, while the number of epochs is set to 100. These parameters are selected to accelerate the training process without compromising accuracy, in addition to ensuring that the model is thoroughly trained without causing overfitting.

In the case of subject-dependent models, the EEGnet is run 10 times, with the data shuffled before each run to utilize different data points for training, validation, and testing for each iteration. To prevent overfitting and ensure a sufficient size of the test data, both the validation set and the test set consist of 25% of the total data points.

4.5 NSGA Settings

The NSGA algorithm requires a high number of runs to achieve optimal results. Given the significant number of variables associated with future selection, 16 variables, and electrode selection, 15 variables, a total of 20 *populations* and 200 *generations* are selected, resulting in a maximum of 4000 runs. Since 4000 runs can be excessive, a tolerance limit of 0.1% is introduced. This makes the algorithm terminate when the objective space changes less than this threshold. Specifically, this criterion is calculated on the last 10 *generations*.

When considering pipeline selection, it becomes evident that running 4000 iterations is excessive, as there are fewer possible combinations of the variables involved. Thus, the *population* number is reduced to 10 and the *generation* number is reduced to 100. Further, the same tolerance limit and calculation criterion are used as for feature and electrode selection.

For feature and electrode selection, *Binary Random Sampling* is employed, while for pipeline selection, *Integer Random Sampling*is used. Binary Random Sampling generates a chromosome consisting of *True* or *False* values, while Integer Random Sampling will produce numbers in between predefined ranges. In addition, the reference direction for all the problems is set to *das-dennis*, as it is widely adopted. Further, *eliminating duplicates* is set to *True* to ensure the integrity of the optimization process.

4.6 Transfer Learning

4.6.1 Deep Learning

TL is a widely employed approach for exploiting earlier knowledge to predict an outcome in a new and similar domain. In the context of DL, TL involves training a pre-existing NN, such as EEGnet, using subjects that exhibit similar EEG characteristics. Subsequently, the network is fine-tuned using a subset of data acquired from the specific subject on which we want to do classifications.

To facilitate the fine-tuning process, each layer of the original network is made untrainable, preserving the weights and biases obtained from the initial training phase. Then an additional trainable layer is introduced to the model, enabling the tuning of parameters specific to the targeted subject. This model is then trained using portions of the subject's data before it is tested using the remaining data.

4.6.2 Machine Learning

Creating a TL model within the framework of ML involves the development of a subject-independent model by utilizing data from multiple subjects for the training procedure. The model can then be utilized for predicting the subject of interest. This underlines how crucial it is to develop a reliable model that can effectively capture the fundamental aspects of the EEG signal generated by various subjects while engaged in MI.

4.6.3 Data splitting

For the subject-independent models developed for the TL-experiments, the data is partitioned a bit differently than for the subject-dependent models. Where 10-fold cross-validation is used for the subject-dependent ML models, the subject-independent models are subject to a different testing methodology. Each subject has 8 independent files of data, thus one of these files per subject is used for testing the TL-model and the remaining for the training of the model. This ensures that the model is trained and evaluated on data from each of the subjects. During the final subject-independent test on an unseen subject, each of the subject's files containing multiple trials is classified separately. Thus, the accuracy, True Positive Rate (TPR), and True Negative Rate (TNR) will be calculated as the average score from these 8 independent runs.

For the subject-independent models implemented using DL and the EEGnet, the data partitioning is again a bit different, as it requires data from the unseen subject to update weights and biases prior to the classification. To build these models, one of the files from each subject is used for testing, one for validation and the remaining for training. When classifying the data from the unseen subject, two of the files are used for testing and the remaining for the tuning of weights and biases. This procedure is repeated 10 times, ensuring a robust evaluation by randomly selecting two files from the target subject for testing during each iteration.

4.7 State Machine for Experimental Test

A state machine serves as a computational model used to represent the current state of the drone and the permissible transitions to subsequent states. For these experiments, the classification of MI tasks is used to facilitate the transition of the drone into different states in real-time, making the drone controllable.

The flow from EEG data stream to the actuation of the drone can be seen in figure 4.6 From this figure, it is shown that an optimal model is made from the database consisting of previously recorded EEG data. This model is further used for the classification of the EEG data stream, which is also one of the files from this database. The optimal model is made based on what is found in the offline experiments. The data processing consists of filtering, optional decomposition, and sequencing of data into 1 second for classification. The stat machines which are used are described in the following subsections.

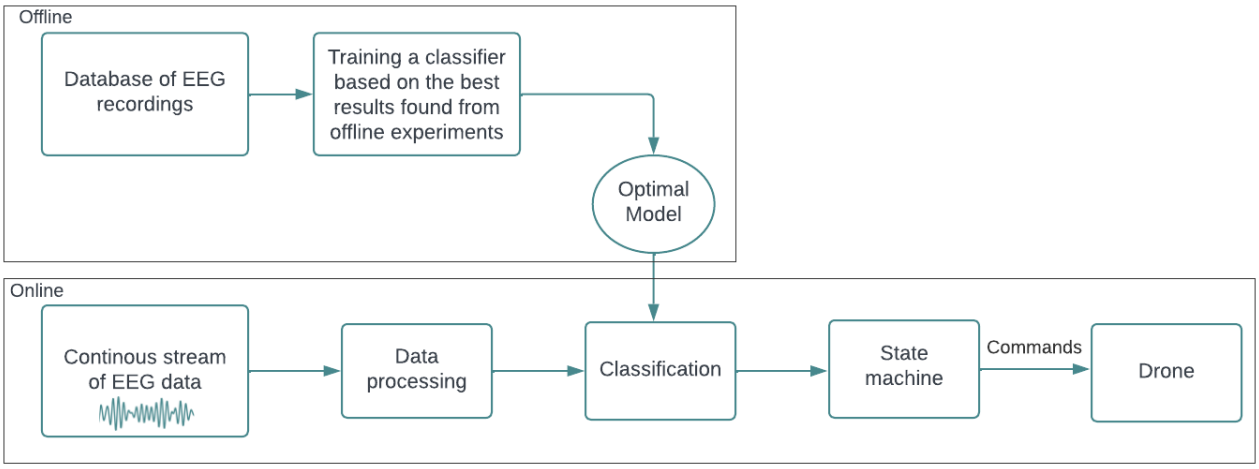


Figure 4.6: Data flow for online classification of EEG data using a stream of data

4.7.1 Drone State Machine for Two-Class Classification

For the two-class MI state machine, the drone is set to either rotate 90 degrees, move forward for 1.5 seconds, or stay idle. Upon initiation, the drone lifts itself from a resting position to a stable flying state. During this initialization phase, all classifications are temporarily disabled until the drone reaches an idle state in mid-air. When in an idle state, the drone awaits a classification output corresponding to either right hand or foot MI. This classification serves as a triggering event, leading the drone to transition to the appropriate subsequent state: rotate or fly forward.

Once the action is completed, the drone reverts to the idle state, where it awaits the next task to perform. As for the initialization, the drone blocks new classifications until it is back to an idle state. The visualization of the state machine can be seen in 4.7.

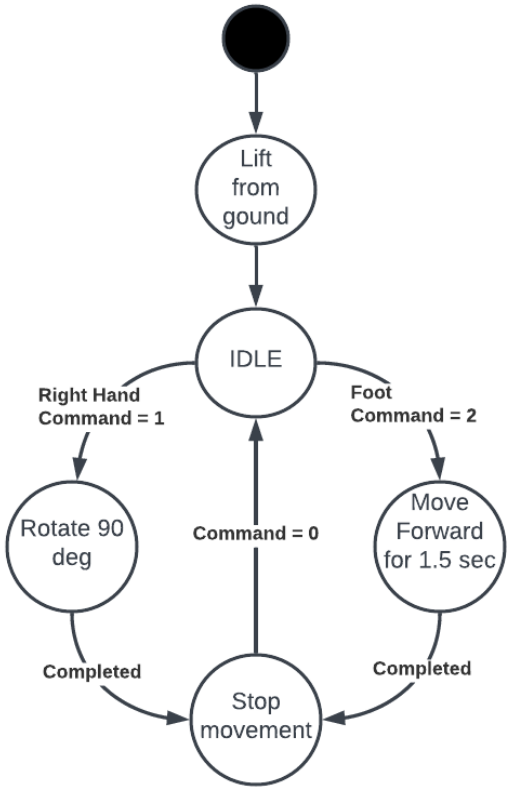


Figure 4.7: State machine of a drone for two MI task classification

4.7.2 Drone State Machine for Three-Class Classification

When rest is added as the third class, the drone gains the capability to autonomously determine when it should return to an idle position. This allows for the continuous segmentation and utilization of a data stream to control the drone's flight. The corresponding state machine for this process is shown in figure 4.8. Similar to the 2-class state machine, the drone blocks all classification during take-off. In addition, to prevent potential collisions, a stopping criterion is added. This way it ensures that if foot MI occurs repeatedly without any rotation of the drone in between, it does not collide with any obstacles.

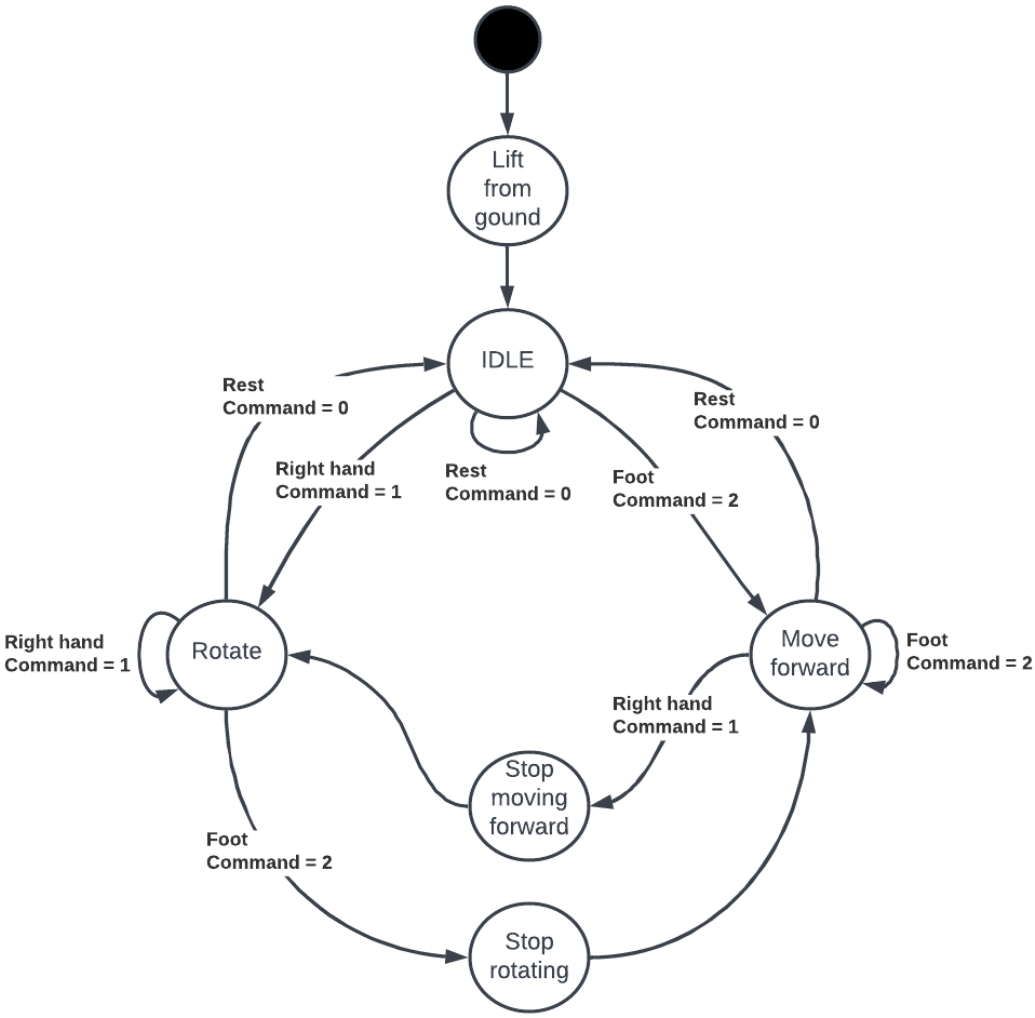


Figure 4.8: State machine for a drone for three classes, including 2 MI tasks and 1 rest task

4.8 Metrics for Evaluation

As the dataset is balanced, accuracy serves as a suitable evaluation metric. Accuracy, denoted as the ratio of correctly classified instances to the total number of instances, provides a measure of the model's overall performance. To ensure reliable accuracy estimation, it is essential to employ separate training and test sets. However, the accuracy metric can be influenced by the specific data partitioning used in these sets. To mitigate this dependency, a k-fold cross-validation technique is employed, unless otherwise specified.

In addition to accuracy, the metrics TPR, also called sensitivity, and False Positive Rate (FPR) is used for performance evaluation. The TPR is the proportion of correctly classified positive instances, while FPR is the proportion of incorrectly classified negative instances. The formulas are defined below:

$$\text{TPR} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{4.1}$$

$$\text{FPR} = \frac{\text{False Negatives}}{\text{False Positives} + \text{True Negatives}} \tag{4.2}$$

Furthermore, the metric specificity is also employed. Specificity, also called TNR, describes the model's ability for correct classification of negatives and can be defined as TNR = 1 − FPR.

Chapter 5

Results

5.1 Subject Selection

For initial testing and to gain an understanding of the dataset's subject performance, a classification of the raw data was conducted. In order to mitigate any potential dependency on the employed feature extraction or classification methods, 5 different pipelines were employed for this initial testing phase. In the following sections of this thesis, more pipelines shall be explored and tested. However, these pipelines will only be tested for a smaller subset of subjects, as it is too time-consuming to test every pipeline on every subject. Which subject and why these were selected will be stated after the initial testing.

Five classification techniques were employed for the initial testing, where each of which involved preprocessing using two different filters, first a notch filter at $50Hz$, then a highpass-filtered at $0.1Hz$. The subsequent techniques are listed below:

- CSP + LDA
- CSP + RF
- DWT + RF
- DWT + LDA
- EEGnet

For the initial tests, all the subbands derived from the DWT were utilized. This includes the 4 levels of decomposition as well as the residual high-frequency coefficient. The results of the initial test are presented in table 5.1. These results indicate significant variation among the subjects and show that the accuracies achieved were relatively low. Further, they show that the different pipelines gave a large range in accuracy, with the lowest recorded accuracy standing at a mere 40.63% and the highest reaching to 89.75%. These extreme values are emphasized in bold within the table. Moreover, the two highest average accuracies are also highlighted in bold. Specifically, these two accuracies belonged to subjects S03 and S09, while the lowest average accuracy was obtained by subjects S06 and S08.

Table 5.1: Classification accuracy for all the subjects in the dataset, where the preprocessing consisted of a notch filtered at $50Hz$ and a highpass filter at $0.1Hz$.

Subject	CSP+LDA	CSP+RF	DWT+LDA	DWT+RF	EEGnet	Average
S01	48.75%	50.63%	56.25%	56.25%	58.25%	54.03%
S02	55.63%	53.75%	55.00%	58.13%	66.00%	57.70%
S03	66.88%	63.75%	85.00%	88.75%	82.00%	**77.28%**
S04	56.25%	51.25%	56.88%	66.25%	68.25%	59.78%
S05	52.50%	55.00%	52.50%	53.75%	64.25%	55.60%
S06	49.38%	56.25%	53.13%	50.00%	54.25%	52.60%
S07	65.63%	61.88%	60.62%	55.00%	69.00%	62.43%
S08	43.75%	**40.63%**	42.75%	48.75%	53.25%	45.83%
S09	75.00%	74.38%	75.63%	78.13%	**89.75%**	**78.58%**
S10	55.00%	53.13%	59.38%	50.00%	54.50%	54.40%
S11	45.00%	43.75%	62.50%	64.38%	50.25%	53.18%
S12	50.00%	50.00%	55.63%	54.38%	55.00%	53.00%
S13	57.50%	55.00%	51.86%	48.13%	53.25%	53.15%
S14	46.25%	56.25%	46.86%	56.25%	58.75%	52.87%

For further initial testing, a bandpass filter ranging from 8 to $30Hz$ was applied to extract the *mu* and *beta* bands, known for exhibiting significant changes in SMR. The bandpass filter was applied preceding the feature extraction using CSP or the signal input to the EEGnet. However, in the case of pipelines using DWT, only the two subbands A_4 and D_4 were extracted instead of bandpass-filtering the signal, as they contain the *mu* and *beta* bands. The results obtained from this experiment are presented in table 5.2. Compared to the previous test, the accuracies were still very variable between the subjects. On the other hand, the overall averages increased for all subjects except subject S09. For instance, subject S08 has a prominent increase of 19.80%. Notably, subjects S03 and S09 achieved the highest accuracies, consistent with the findings from the prior test without bandpass filtering or subband removal. Further, comparing all accuracies for all the pipelines, the lowest accuracy obtained was 46.88% and the highest was 99.25%, both marked with bold text in the table.

Table 5.2: Classification performance for all the subjects in the dataset, where the preprocessing consisted of either bandpass filtering or extraction of 2 DWT levels.

Subject	CSP+LDA	CSP+RF	DWT+LDA	DWT+RF	EEGnet	Average
S01	65.00%	60.63%	61.88%	50.00%	52.25%	57.95%
S02	66.88%	61.88%	**46.88%**	49.38%	78.50%	60.70%
S03	97.50%	95.63%	87.50%	86.25%	**99.25%**	**93.23**%
S04	80.63%	78.13%	66.88%	70.63%	86.50%	76.55%
S05	66.25%	66.25%	60.00%	50.00%	58.00%	60.10%
S06	75.00%	70.63%	51.25%	53.13%	81.50%	66.30%
S07	73.75%	74.38%	62.50%	56.88%	81.50%	69.80%
S08	83.13%	86.88%	59.38%	48.75%	50.00%	65.63%
S09	69.38%	83.75%	66.88%	79.38%	92.75%	**78.43**%
S10	61.88%	54.38%	58.13%	54.37%	68.25%	59.40%
S11	76.88%	75.00%	58.75%	63.13%	86.00%	71.95%
S12	71.88%	68.13%	55.00%	51.23%	53.00%	59.85%
S13	59.38%	55.00%	51.25%	52.50%	50.25%	53.68%
S14	59.38%	54.38%	48.13%	53.75%	55.50%	54.23%

For the subsequent sections of the thesis, four subjects were chosen for further experimentation and model improvement. The subjects chosen were S03 and S09, as they achieved the highest overall accuracies. Additionally, subjects S11 and S13 were included in the selection. Subject S11 was specifically chosen due to its initially low average accuracy of 53.18% during the first test, which increased to 71.95% by simply applying a bandpass filter and removing bands from the DWT. It is therefore of great interest to investigate whether an optimized pipeline can yield further improvement for this subject. On the other hand, subject S13 was selected because it was the lowest-performing subject after the second initial test. Similar to subject S11, it is interesting to investigate the potential for accuracy improvement through further optimization.

The upcoming section in this thesis will use the two subbands A_4 and D_4 of the DWT decomposition unless stated otherwise. This decision is based on the findings from the initial tests, where it was observed that the overall accuracies for the DWT-based pipelines increased when employing only two subbands instead of all 5. Moreover, utilizing two bands would also lead to a reduction in computational complexity.

5.2 Feature Selection using NSGA

Calculating a lot of features is both computationally expensive and time-consuming. The objective of this section was thus to use the NSGA algorithm to reduce the number of features while maintaining the highest possible accuracy. The selection of features was done based on the accuracies obtained from the algorithm. For the highest recorded accuracies, an investigation was carried out to determine the number of and which specific features were employed to achieve these scores. This analysis was conducted for all four subjects and across 8 different pipelines. Utilizing different pipelines ensured that the features chosen were not dependent on the pipeline itself, as the most important features may vary dependent on the pipeline.

The chance level of a two-class classification is conventionally defined as 50%. However, the accuracies obtained in initial tests were observed to be as low as 40.63%, falling below the chance level. This suggests that the classifier is more biased towards one of the classes. While such outcomes might be expected in an unbalanced data set, it is worth noting that the dataset utilized for this thesis is balanced, and 10-fold cross-validation was implemented to reduce the chance of this occurring. Thus, the reason behind these results remains unknown and has not been further investigated. Consequently, to account for these findings, the chance level is adjusted to 60% as it represents a threshold that can be obtained purely by chance.

The manual feature extraction process involved the application of two distinct methods: bandpass filtering with a range of $8Hz$ to $30Hz$ and DWT subband extraction. When bandpass filtering was employed, the features were computed based on the entire frequency band. For subband extraction with DWT, the features were calculated for each individual subband, which in this case were A_4 and D_4.

To determine the optimal number of features to use, a count for each subject was made using the results from the NSGA algorithm. The recorded results consisted of the accuracies achieved and the corresponding feature subset for each classification done. Upon inspection of the results from the different subjects, it became evident that the threshold representing what was a high score varied a lot. This laid the foundation for defining four different thresholds which categorized high accuracy levels for each subject. Specifically, for subject S03, the threshold was set to 90.00%, while for subject S09, it was 80.00%. Further, for subject S11m the threshold was defined as 65.00%, and for subject s13, it was set to 60.00%. Based on these established thresholds, the number of times each feature was used to obtain higher accuracies than the thresholds for each subject was counted. This count can be seen in table 5.3.

From this table, a pattern is revealed. None of the subjects employed either 1, 15, or 16 features to attain accuracies higher than the established thresholds. Furthermore, the count increases until it reaches 4 or 5 features, after which it gradually declines. This pattern is further visually represented in the bar chart shown in figure 5.1. The bar chart sums up the counts across the subjects, showing that the NSGA algorithm most often obtained high accuracies when choosing 4 to 6 features. Upon further examination of the accuracies achieved using 4, 5, or 6 features, the difference in accuracy was relatively small. Based on these results, in addition to considering the computational complexity and time a consuming aspect of feature computation, the optimal number of features was set to 4 features.

Table 5.3: Count of how many features used to obtain accuracies above the given thresholds, 90% for S03, 80% for S09, 65% for S11, and 60% for S13.

Subject	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
S03	0	18	254	614	786	644	504	377	244	106	31	12	1	0	0	0
S09	0	14	317	580	772	586	364	153	77	46	16	15	0	1	0	0
S11	0	23	276	641	572	450	290	117	64	33	10	11	0	0	0	0
S13	0	3	38	135	144	67	18	10	5	2	1	0	0	0	0	0
Sum	0	58	885	1970	2274	1747	1176	657	390	187	58	38	1	1	0	0

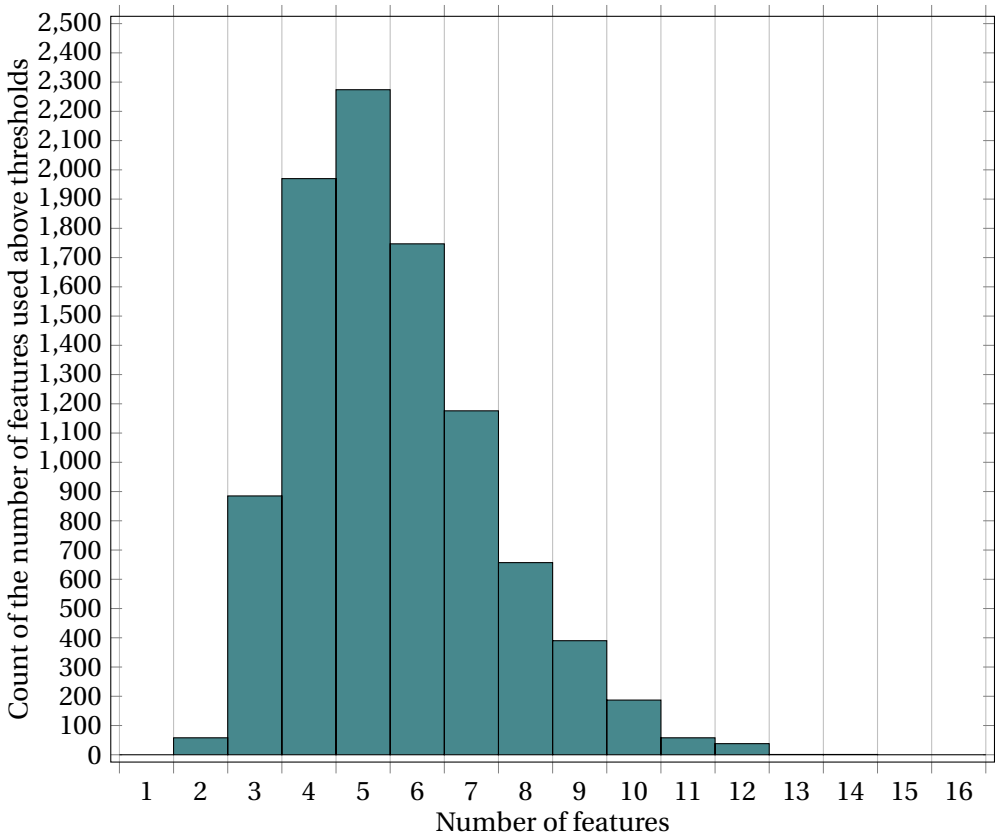


Figure 5.1: Bar chart showing the count of the number of times a certain number of features are used to obtain accuracies above given thresholds for all the subjects combined.

In order to identify the four features that significantly contributed to achieving accuracies above the chosen thresholds, the results of the NSGA algorithm were used to make a heat map for each subject. The heat maps display a count of how many times each feature was selected for each of the models. The resulting heat maps are shown in 5.2, 5.3, 5.4 and 5.5.

5.2.1 Feature Selection for Subject S03

For subject S03 the results obtained for feature selection are displayed in figure 5.2. We can see that the four features used most times to obtain high accuracies were TE, LL, Mean, and Skewness.

Features\Methods	BP SVM	BP GB	BP RF	DWT SVM	BP LDA	DWT RF	DWT GB	DWT LDA	Sum
Teager Energy	1099	575	331	131	269	0	0	4	2409
Line Length	887	215	311	119	122	2	5	2	1663
Mean	1137	203	77	17	79	16	1	1	1531
Skewness	1117	64	75	129	42	0	0	2	1429
Standard Deviation	602	247	86	289	72	9	5	5	1315
Higuchi Fractal Dimension	845	129	66	116	54	0	1	0	1211
Katz Fractal Dimension	859	124	77	29	34	1	0	3	1127
Variance	558	137	92	114	123	13	5	3	1045
Hjorth Mobility	503	83	187	85	152	4	10	0	1024
Petrosian Fractal Dimension	486	177	49	101	123	8	5	4	953
Instantaneous Energy	332	228	128	109	110	7	10	5	929
Root Mean Square	297	108	61	79	91	12	16	3	667
Hjorth Complexity	173	92	57	233	53	23	22	0	653
Peak To Peak	375	159	61	0	45	3	1	0	644
Kurtosis	311	128	99	48	46	1	0	1	634
Sevcik Fractal Dimension	146	187	48	107	41	0	7	3	539
Sum	9727	2856	1805	1706	1456	99	88	36	

Figure 5.2: Count of how many times the NSGA algorithm selects each feature to obtain an accuracy above 90% for subject S03.

5.2.2 Feature Selection for Subject S09

The results obtained for subject S09 showed few similarities to the ones obtained for S03, but LL is one of the four best features for both subjects. From table 5.3 it can be seen that the four most prominent features are LL, Kurtosis, SD, and IE.

Features\Methods	BP GB	BP RF	DWT GB	BP SVM	DWT RF	BP LDA	DWT SVM	DWT LDA	Sum
Line Length	647	203	595	241	273	37	28	0	2024
Kurtosis	597	107	188	323	45	4	0	0	1264
Standard Deviation	184	349	161	210	110	19	10	0	1043
Instantaneous Energy	155	331	167	131	78	18	9	0	889
Sevcik Fractal Dimension	363	220	111	99	87	6	0	0	886
Teager Energy	253	251	156	86	107	5	25	0	883
Variance	175	154	162	197	68	6	16	0	778
Root Mean Square	109	325	154	106	43	16	13	0	766
Hjorth Mobility	256	157	78	96	125	2	2	0	716
Hjorth Complexity	205	78	301	30	71	1	0	0	686
Higuchi Fractal Dimension	255	99	85	113	45	4	0	0	601
Katz Fractal Dimension	132	103	194	74	57	6	1	0	567
Petrosian Fractal Dimension	260	100	70	42	57	1	2	0	532
Mean	169	94	63	66	63	0	15	0	470
Skewness	89	80	55	65	19	11	0	0	319
Sum	3849	2651	2540	1879	1248	136	121	0	

Figure 5.3: Count of how many times the NSGA algorithm selects each feature to obtain an accuracy above 80% for subject S09.

5.2.3 Feature Selection for Subject S11

Table 5.4 displays the results obtained by running feature selection on subject S11. As for S09 LL, Kurtosis, and SD were among the three of the four best features, but the order of the features differed. Further, subject S11 had HMO as the second most prominent feature which did not appear among the four best features for S09.

Features \Methods	DWT GB	DWT RF	BP RF	BP SVM	BP GB	BP LDA	DWT SVM	DWT LDA	Sum
Line Length	767	475	147	73	58	39	28	4	1591
Hjorth Mobility	242	956	55	42	11	9	1	1	1317
Kurtosis	524	279	29	55	8	8	0	0	903
Standard Deviation	94	69	262	164	84	71	34	2	780
Petrosian Fractal Dimension	145	101	117	145	118	116	3	2	747
Mean	234	203	102	39	82	28	40	7	735
Instantaneous Energy	151	110	135	132	76	62	15	0	681
Higuchi Fractal Dimension	123	104	160	35	159	11	23	3	618
Peak To Peak	222	248	47	45	15	7	20	1	605
Variance	88	52	205	79	90	67	20	3	604
Teager Energy	206	82	63	76	65	50	33	3	578
Root Mean Square	146	74	143	77	58	59	10	2	569
Skewness	65	244	100	26	58	21	0	0	514
Katz Fractal Dimension	146	204	37	19	65	4	0	2	477
Sevcik Fractal Dimension	259	71	26	51	36	15	1	0	459
Hjorth Complexity	117	43	44	53	16	20	13	2	308
Sum	3529	3315	1672	1111	999	587	241	32	

Figure 5.4: Count of how many times the NSGA algorithm selects each feature to obtain an accuracy above 65% for subject S11.

5.2.4 Feature Selection for Subject S13

Subject S13 was the lowest performing subject, and as we can see from table 5.5 the number of times the features appear was overall much lower than for the other subjects. The table indicates that the four most prominent features were Skewness, HMO, HFD, and Kurtosis.

Features\Methods	BP SVM	BP LDA	DWT LDA	DWT RF	DWT SVM	BP RF	BP GB	DWT GB	Sum
Skewness	239	103	9	5	1	1	4	0	362
Hjorth Mobility	216	28	3	26	8	16	1	0	298
Higuchi Fractal Dimension	159	32	13	4	22	4	0	0	234
Kurtosis	79	76	19	0	6	0	4	0	184
Petrosian Fractal Dimension	97	11	19	9	28	0	1	0	165
Instantaneous Energy	69	26	18	11	7	10	5	0	146
Mean	36	8	21	46	25	0	0	0	136
Sevcik Fractal Dimension	99	20	10	2	0	0	1	0	132
Teager Energy	48	17	16	10	28	7	0	0	126
Standard Deviation	35	45	13	4	5	6	10	0	118
Root Mean Square	37	41	3	8	3	7	8	0	107
Peak To Peak	37	33	7	11	10	0	2	0	100
Katz Fractal Dimension	16	50	17	12	1	0	0	0	96
Hjorth Complexity	40	3	11	24	4	1	0	0	83
Line Length	21	9	18	12	12	3	4	0	79
Variance	15	11	14	3	5	10	10	0	68
Sum	1243	513	211	187	165	65	50	0	

Figure 5.5: Count of how many times the NSGA algorithm chooses each feature to obtain an accuracy above 60% for subject S13.

The analysis of the results revealed certain similarities among the subjects, but very few. Notably, the features LL, Kurtosis, Skewness, HMO, and SD appeared among several subjects. Specifically, LL and Kurtosis were chosen for three out of four subjects, while Skewness, HMO, and SD were chosen for two. Further, the remaining chosen features were subject-dependent, thus appearing only for one of the four subjects.

5.3 Pipeline Selection using NSGA

Due to many pipelines, the problem of identifying the optimal pipeline for each specific subject arises. Addressing this problem, the NSGA algorithm was used, optimizing on the highest accuracy score. This way, we were able to reduce the number of pipelines for further experiments, while also finding which pipelines performed well across subjects. As a feature selection already was conducted, the resulting features for each subject were used in this optimization problem. Furthermore, our previous findings showed that extracting two of the subbands from the DWT yielded higher results than using all. Hence, this approach was adopted using only the two relevant frequency bands in subsequent analyses.

Since the different pipelines consist of various combinations of filtering, feature extraction, number of CSP features, and classification algorithm, the pipelines can be broken down into these four points:

- Frequency band: *all, mu, mu and beta (mu-beta), beta.*
- Number of CSP-features: *3-7.*
- Feature extraction method: *DWT, Bandpass, CSP.*
- Classifier: *EEGnet, RF, GB, LDA, SVM.*

Not all the pipelines require all the points above, thus some conditions were added to the algorithm:

- *If DWT is chosen as the feature extraction technique, the frequency band used was set to "all", making sure the signals were not filtered before the wanted DWT bands were extracted.*
- *The number of CSP features was only used when CSP was chosen as the feature extraction method.*
- *When EEGnet was chosen as the classifier, no feature extraction method was used.*

In total combining the feature extraction methods, frequency bands, number of CSP features, and classifiers, with the conditions as a baseline gave a total of 104 unique pipelines.

5.3.1 Pipeline Selection for Subject S03

During the optimization process targeting subject S03, the highest accuracy score obtained was 100.00%. This was achieved using the EEGnet architecture where the preprocessing consisted of a mu, mu-beta, or beta bandpass filter. Despite getting the same accuracy across all three bands, a decision was made to proceed using the mu-beta for further investigations. This choice was made as the SMR changes are said to appear in this band and it was anticipated that the use of a wider frequency band may aid in reducing the number of electrodes required in subsequent experiments.

An accuracy of 99.38% was obtained when employing both SVM and LDA as classifiers. In both pipelines, the beta band was utilized in combination with CSP as the feature extraction method. The SVM classifier obtained this accuracy performance on two occasions, employing both 3 and 4 CSP-features, while LDA obtained it once using 7 CSP-features. Further, subject S03 obtained an accuracy of 98.75% when employing LDA, with the beta band, and CSP with 3, 4, and 6 features. This observation potentially suggests that the combination of LDA with CSP as a feature extraction method consistently yields high accuracy values on average, which is interesting to further investigate in the electrode selection. Consequently, the LDA classifier in combination with the beta band and CSP with 7 features was selected for subsequent experiments.

5.3.2 Pipeline Selection for Subject S09

Similar to the observations made for subject S03, subject S09 also achieved the highest accuracies when employing the EEGnet. Notably, the highest obtained accuracy was 93.80%. However, the only difference was that S09 obtained this value using all the different frequency bands. Again, as the change in SMR is said to appear in the mu-beta band, and with the hope that using a larger band might facilitate a reduction of the number of electrodes, we continued using EEGnet with the mu-beta frequency band in further experiments.

The next best classifiers obtained 10% lower accuracy than the EEGnet, giving an accuracy of 83.75%. Specifically, this accuracy was obtained utilizing two different pipelines: CSP with RF and Bandpass feature extraction with GB, both with the employment of the mu-beta frequency band. Moreover, utilizing CSP with RF additionally obtained accuracies of 83.13%, 82.50%, and 81.88% with varying numbers of CSP features. Based on these results, the CSP with RF pipeline was chosen for further experimentation using 3 CSP features.

5.3.3 Pipeline Selection for Subject S11

In the case of subject S11, the highest accuracy obtained was 87.5%. This was achieved utilizing the EEGnet classifier, employing both mu and mu-beta frequency bands. Making the same conclusion as with S03 and S09, the decision was made to use the mu-beta band for further experiments.

The second-best accuracy score of subject S11 was 77.50% when employing CSP with 5 features, the mu-beta frequency band, and RF classifier. Furthermore, an accuracy of 76.50% was obtained using CSP with 4 features, the mu frequency band and LDA, while an accuracy of 75.63% was achieved using CSP with 6 features, the mu-beta band and RF. Given these results, we continued to explore the combination of CSP with 5 features, RF, and the mu-beta band.

5.3.4 Pipeline Selection for Subject S13

In line with the trend observed across the other subjects, subject S13 obtained the highest accuracy score of 68.60% using EEGnet classifier. On the other hand, S13 obtained this result using just the mu frequency band, which was thus used in the subsequent experiments for this subject.

Further, the results showed that employing CSP with 7 features, in combination with the mu-beta frequency band and RF classifier, yielded an accuracy score of 68.13%. The following best classifier gave an even lower accuracy score, yielding a score of 64.38%. Thus, for further experiments, the pipeline involving CSP with 7 features, the mu-beta band, and RF classifier was selected for further investigation.

5.3.5 Results from Pipeline Selection Across Subjects

As seen from these results, it is apparent that the performance of the DWT feature extraction method falls short in comparison to the other employed techniques. Similar observations can be made for SVM and GB, which do not appear among the best-performing classifiers. Notably, the results showed that the EEGnet pipeline consistently outperforms all the other pipelines, but not always by a high percentage. Additionally, a trend seems to be that using CSP with RF yields high accuracy scores. This trend is consistent across all subjects, except for subject S03. However, on close examination of the results obtained from subject S03, it becomes evident that the CSP-RF pipeline gives 97.50% accuracy using 3, 4, and 5 CSP features, which was not far from the highest accuracy score of 100.00%.

The findings presented in this section show that, overall, the CSP feature extraction method shows better performance when compared to both the DWT and Bandpass techniques. On the other hand, from the results, it seems to be few correlations on how many CSP features are needed to obtain high accuracy scores. This suggests that the influence of the number of CSP features on accuracy is not straightforward, requiring further investigations. Moreover, the results showed that the EEGnet pipeline consistently emerges as the optimal choice, both for high- and low-performing subjects, reaffirming that it is the best pipeline for EEG classification tasks.

5.4 Rereferencing versus Electrode Selection

The optimal pipelines found during the pipeline selection process were subjected to additional analysis involving electrode selection and the application of the Laplacian rereferencing system. The purpose of this analysis was to investigate if a reduction in the number of electrodes could enhance the classification models, and to reduce the complexity of the models. The optimal pipelines found in the preceding section are presented below:

- S03:
  - bands: mu-beta, classifier: EEGnet
  - bands: beta, CSP with 7 levels, classifier: LDA
- S09:
  - bands: mu-beta, classifier: EEGnet
  - bands: mu-beta,CSP with 3 levels, classifier: RF
- S11:
  - bands: mu-beta, classifier: EEEGnet
  - bands: mu-beta, CSP with 5 levels, classifier: RF
- S13
  - bands: mu, classifier: EEGnet
  - bands: mu-beta, CSP with 7 levels, classifier: RF

5.4.1 Laplacian Rereferencing

Upon employing Laplacian referencing combined with the pipelines above, the results displayed in table 5.4 were obtained. From this table, it is evident that there were no significant changes in the accuracies obtained for subject S03. Specifically, the utilization of the EEGnet pipeline yielded the same results, while a small decrease in accuracy was observed when using the pipeline with the LDA classifier.

For subject S09, a small decrease in accuracy was observed when the EEGnet pipeline was employed, while a more drastic decrease was observed when utilizing the RF pipeline. This trend seems to be persistent for the remaining two subjects as well, regardless of the pipeline used. Notably, the results obtained, excluding S03, were mainly characterized by a drop in accuracy when introducing Laplacian rereferencing to the pipelines. For example, subject S09 experienced a drop in accuracy from 83.75% to 68.75%, giving a decline of 15.00% when introducing the laplacian rereferencing to the pipeline based on CSP and RF.

It is worth noticing the high SD associated with the pipeline CSP with RF, indicating significant variability in accuracy scores across different runs. For instance, in the case of subject S09, this variability can result in performances ranging from relatively high to as low as random guessing. Furthermore, the results clearly indicate that the introduction of the Laplacian rereferencing scheme does not yield improvements in accuracy or enhance the models. Based on these results, the Laplacian rereferencing was excluded from further experiments.

Table 5.4: Classification performance using Laplacian referencing scheme and different classification methods. For all subjects except for S13, the EEGnet was run with both mu-beta bands, for S13 only the mu band was used. Further, for S03, CSP with 7 features classified by LDA was used. For the remaining subjects, CSP RF was used, and the number of CSP features used was 3, 5, and 7 for S09, S11, and S13 respectively.

Subject	CSP+LDA	CSP+RF	EEGnet
S03	94.38% ± 4.61%	-	99.50% ± 1.05%
S09	-	68.75% ± 11.79%	87.75% ± 5.33%
S11	-	63.75% ± 10.12%	69.25% ± 7.17%
S13	-	47.50% ± 16.72%	45.75%± 3.55%

5.4.2 Electrode Selection

Due to the noticeable decrease in accuracies observed using Laplacian rereferencing, an alternative method for reducing the number of electrodes was explored. In this experiment, the NSGA algorithm was employed with the objective of achieving a high accuracy score while simultaneously minimizing the number of electrodes.

Figure 5.6 show the numbering scheme assigned to the electrodes used for recording purposes, ranging from 1 to 15. Moreover, the center electrodes are placed according to the 10-20 system, thus having the names C3, Cz, and C4. As earlier explained, the signal associated with foot MI tasks is often located at the top of the head, aligning with electrodes 2, 7, 8 (Cz), 9, and 14. Further, right hand MI-signals occur mostly on the left side of the brain, corresponding to electrodes 1, 4, 5 (C3), 6, and 13.

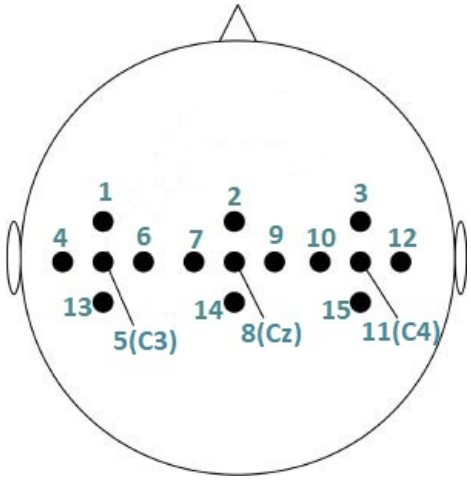


Figure 5.6: Numbering of the electrodes used for recording the EEG signals. Adapted from [67].

Electrode Selection for Subject S03

As previously established, subject S03 consistently demonstrates good performance, which was also reflected in the results obtained in the electrode selection. When using the EEGnet, this subject obtained 100.00% accuracy with multiple combinations of electrodes. The number of electrodes associated with 100.00% accuracy varied from 2 to 14. Notably, when 2 electrodes were used, their placement consistently occurred either in the left or middle region of the brain. In the cases involving 3 electrodes, the electrodes were located in the left or middle part of the head most of the time, but occasionally with 1 electrode positioned on the right side. For scenarios involving more than 3 electrodes and 100.00% accuracy, the selection sometimes included electrodes on the right side, but never more than 2 electrodes, and always with a greater portion from the left or middle regions.

During the execution of electrode selection utilizing CSP and RF we also obtained an accuracy of 100.00%. However, the number of electrodes required to obtain this accuracy was higher compared to the EEGnet. The lowest number of electrodes corresponding to 100.00% accuracy was 6, using electrodes 4, 5 (C3), 6, 7, 8 (Cz), and 10. Electrode number 10 is situated on the right side of the head, but is the furthest left electrode in that particular section. As the number of electrodes used to achieve 100.00% accuracy increased, the occurrence of right-side electrodes was more frequent. Nevertheless, the largest proportion of electrodes was located within the left and middle sections.

Upon reducing the number of electrodes to 2, 3, 4, and 5, the accuracy decreased to 91.88%, 96.87%, 98.75%, and 99.38% respectively. These accuracies were obtained using electrodes positioned within the left and the middle section alone. In general, the results showed that whereby the inclusion of right-side electrodes was infrequent when few electrodes were used to obtain high accuracies.

The findings from both the EEGnet and CSP-LDA pipelines revealed that using one single electrode resulted in a big decrease in accuracy or random guessing. Furthermore, when electrode selection was done using CSP-LDA, using 2 electrodes yielded suboptimal and low accuracies, in contrast to EEGnet. Additionally, a trend found was that low accuracies or random guessing tended to occur more frequently when a larger portion of electrodes was located in the right section of the head, regardless of the method employed.

Electrode Selection for Subject S09

Subject S09 obtained its highest accuracy of 93.80% using the EEGnet. In most instances, this accuracy was obtained using 2, 3, or 4 electrodes, although occasionally a higher number of electrodes was also utilized. When using 2 electrodes, their location consistently occurred within the left and middle sections. However, for 3 and 4 electrodes, there was occasionally one electrode from the right section. As in the case of subject S03, a higher number of electrodes correlated with a more frequent occurrence of electrodes from the right side. Further, low accuracies occurred when only 1 electrode was used, or when 2 electrodes were used, with 1 or 2 of them positioned in the right section of the head. In addition, low accuracies were occasionally observed using a higher number of electrodes. In that case, the biggest portion of electrodes was located on the right side.

When using CSP and RF, a higher number of electrodes was often needed to obtain high accuracies. The highest accuracy obtained was 87.50%, which required the use of 9 electrodes located mostly within the left and middle sections of the head. Reducing the number of electrodes resulted in a small reduction in accuracy, although not significantly. Notably, an accuracy of 86.25% was obtained using only 2 electrodes, specifically electrodes 8 and 13. Overall, the analysis showed that with high accuracies above 80.00% and the use of few electrodes, their location was in most cases located within the left and middle sections of the head. When the classification was done using only 3 or 4 electrodes, and the electrodes were placed either close to each other or at least one electrode placed on the right side, low accuracies occurred. Furthermore, the classification using a single electrode resulted in random guessing.

Electrode Selection for Subject S11

Subject S11 obtained its highest accuracy of 93.80% when using EEGnet with the 3 electrodes 3, 6, and 11 (C4). Out of this selection, 2 of them are located within the right section of the head. In fact, a general trend from this subject showed that at least 1 of the electrodes used when high accuracies were obtained was located in the right section of the head. In addition, a few of the electrodes tended to be in the middle section. This led to the general trend indicating that accuracies were often achieved by employing electrodes from both the left and right sections of the head. As the number of electrodes increased, the accuracy decreased and the selection of electrodes was from all 3 sections.

As for subjects S03 and S09, using 1 electrode resulted in low accuracies or random guessing, where the highest obtained accuracy with one electrode was 65.60%, using electrode number 11. In addition, random guessing was also observed when the selection of electrodes was higher than 1, but unlike S03 and S09, the occurrence of right-section electrodes was not the crucial factor associated with the low accuracies.

For the context of ML-based electrode selection, it was observed that the highest accuracies obtained, ranging from 77.50% and 79.38%, required the use of a high number of electrodes, specifically between 6 and 10. A general observation was that the electrodes chosen were predominantly located within the middle and right sections, except electrode number 6, which is the rightmost electrode in the left section. The highest accuracy was obtained using 10 electrodes, with 4 of them located within the right section. Upon reducing the number of electrodes to the range of 3 to 5, the accuracy decreased to around 75.00%. Even in these reduced selections, the chosen electrodes continued to originate from the right and middle sections, along with the inclusion of electrode 6.

Using 1 and 2 electrodes frequently resulted in outcomes close to random guessing. For the case of 2 electrodes, the selection of electrodes tended to be placed close to each other, as exemplified by the pairing of electrode numbers 8 (Cz) and 14. Furthermore, a score between 55.00% and 60.00% was often achieved using 3 electrodes, without any visible pattern on which electrodes were chosen.

Electrode Selection on Subject S13

Subject S13 is by far the worst-performing subject; however, the implementation of electrode selection yielded an enhancement in the performance to some extent. The highest accuracy of 75.00% was obtained with the EEGnet in combination with 4 electrodes. This was achieved with 3 different electrode combinations, where all, except electrode number 3, were located in the left and middle sections of the head. By reducing the number of electrodes to 3 and 2, the accuracy decreased to 68.80% and 71.89% respectively. When employing 3 electrodes, electrode number 3 was again used to obtain the highest accuracy. This was also a pattern that persisted when involving a higher number of electrodes in the selection. Employing a range of 5 to 10 electrodes yielded the highest accuracy of 71.89%, all using electrode number 3. On the other hand, the remaining electrodes were primarily from the left and middle sections, a characteristic shared with the selection of 2 electrodes.

Random guessing and low accuracies were mostly obtained using 1 electrode, but they also emerged when 2 electrodes were used. In the case of 2 electrodes, the selection of electrodes frequently comprised electrodes from the same section or one from the left section in combination with another from the right section. Given that subject S13 is a bad-performing subject, a high number of the obtained accuracies fell below the 60% threshold using all variations of electrode selection. Consequently, it was challenging to find any further patterns contributing specifically to low performance.

In the case of ML, the highest obtained accuracy was 68.75%. This was obtained using 9 electrodes, with 4 of them located in the right section of the head. Similarly, other high accuracies were obtained when using a high number of electrodes, yet there were non or close to no patterns to find. Reducing the number of electrodes led to a decrease in accuracy, yielding a performance of 65.00%. As for the higher number of electrodes, no pattern emerged regarding the selection of electrodes that yielded high accuracies. Further, in accordance with high accuracies, few patterns became apparent of which electrodes gave low performance or random guessing, aside from the one observation being that few electrodes yielded low results.

General Observations of Electrode Selection

For all subjects, it was observed that achieving high accuracies using ML required a higher number of electrodes than for DL. In addition, it was also seen that the increase in accuracy was highest on DL, and it obtained better scores than ML.

Regarding the specific electrode selections for attaining high accuracies, few patterns were observed from the results. For subjects S03, S09, and S13, utilizing DL-techniques showed that high accuracies were frequently associated with a larger number of electrodes positioned within the left and middle section of the head. This observation aligns with the theory saying that foot and right hand MI occurs in these regions. On the other hand, subject S11 seemed to need electrodes from the right side of the head to obtain high accuracies. Further, for subjects with lower overall performance such as S13, finding any specific pattern yielding high accuracies proved to be particularly challenging.

5.4.3 CAR Rereferencing

The results obtained from the application of Laplacian rereferencing and electrode selection showed variations in performance. Specifically, utilizing electrode selection resulted in higher accuracies, whereas the inclusion of Laplacian rereferencing decreased the accuracies. Another rereferencing method, known as the CAR method, offers an alternative approach. Unlike electrode selection and Laplace rereferencing, CAR does not reduce the number of electrodes but instead employs the common average of the electrodes to rereference each electrode.

In this section, the CAR method was applied to investigate if it could further enhance the performances. To achieve this, the CAR method was applied to the optimal pipelines determined in pipeline selection, in conjunction with the outcome of the electrode selection process. As the objective of electrode selection was to reduce the number of electrodes utilized while obtaining a high accuracy, the trade-off between accuracy and the number of electrodes was closely examined to find which selection to further use. The resulting pipelines and electrodes used in this section were thus as follows:

- S03
  - Band: mu-beta, Classifier: EEGnet, Electrodes: 5 (C3) and 14.
  - Band: beta, Feature extraction method: CSP with 7 features, Classifier LDA, Electrodes: 4, 5, 7, 8 (Cz).
- S09
  - Band: mu-beta, Classifier: EEGnet, Electrodes: 2 and 13.
  - Band: mu-beta, Feature extraction method: CSP with 3 features, Classifier: RF, Electrodes: 8 (Cz) and 13.
- S11
  - Band: mu-beta, Classifier: EEGnet, Electrodes: 4, 6, and 11 (C4).
  - Band: mu-beta, Feature extraction method: CSP with 5 features, Classifier: RF, Electrodes: 2, 6, 7, 8 (Cz), 11 (C4), and 14.
- S13
  - Band: mu, Classifier: EEGnet, Electrodes 2, 3, 4, and 14.
  - Band: mu-beta, Feature extraction method: CSP with 7 features, Classifier: RF, Electrodes 9, 10, 11 (C4), 13, and 14.

The obtained results for the classification experiment with and without CAR rereferencing are presented in table 5.5. These results show that the accuracies obtained when applying the CAR rereferecing decrease for the DL method EEGnet. Contrarily, the accuracies for the ML methods remain unchanged. This observation indicates that introducing the CAR method might be excessive for the ML pipelines and have an adverse effect on the DL pipelines. Therefore, the CAR rereferencing will be excluded from further experiments.

Table 5.5: Results obtained with and without CAR rereferencing. Classification methods used are EEGnet for all subjects, and CSP + RF for all subjects except S03. S03 uses CSP + LDA.

Subject	rereferencing	EEGnet (DL)	Machine Learning
S03	without CAR	100.00% ±0.00%	98.75% ± 3.95%
	CAR	99.50% ± 1.05%	98.75% ± 3.95%
S09	Without CAR	93.80% ± 3.17%	86.25% ± 8.22%
	CAR	91.50% ± 3.57%	86.25% ± 8.22%
S11	Without CAR	93.80% ± 4.50%	77.50% ± 11.86%
	CAR	80.75% ± 6.35%	77.50%± 11.86%
S13	Without CAR	75.00% ± 5.77%	66.88% ± 14.75%
	CAR	52.75% ± 6.66%	66.88%± 14.75%

5.5 Three-class motor imagery classification

In order to ensure reliable control of a drone using MI signals, it is crucial to prevent the model from classifying MI when it is not present. If the model were to incorrectly identify MI when it is absent, the behavior of the drone would become unpredictable and potentially unsafe. To address this concern, the introduction of a third class, known as the "rest" class, becomes necessary. Consequently, the model is extended to classify three distinct classes: 1) hand, 2) foot, and 3) rest.

To conduct these experiments, both the complete set of electrodes and the optimal set of electrodes obtained through the electrode selection on two classes were tested. From the pipeline selection, it was found that the EEGnet was the most optimal classifier for all subjects, thus it was used in this experiment. Additionally, the CSP with RF emerged as the next best pipeline for all subjects except S03. Since both the CSP with RF and CSP with LDA gave the same accuracy for S03 and as we wanted to reduce the number of pipelines such that a more general model could be found, CSP with RF was used for all subjects.

The outcomes of the 3-class classification employing both the DL and ML pipelines for all subjects are presented in table 5.6. Analysis of these results of the DL pipeline revealed a notable decrease in accuracies across subjects when utilizing the optimal electrodes compared to all electrodes, except for subject S13. Even though S13 slightly increased, it was still close to random guessing. Further, when employing the ML pipeline, the accuracies decreased for S03 and S13, while increasing for S09 and S11 when using the selection of the optimal electrodes. Nevertheless, the decrease in accuracy for subject S13 is relatively small, making it less significant compared to the other results.

Further, subject S03 obtained the highest accuracies, consistent with expectations. Despite its high accuracy, there was a small drop in accuracy compared to the 2-class classification. For the remaining subjects, a clear drop in accuracies from previous experiments was observed, where the ML pipelines gave an especially bad performance. In particular, subject S13 obtained around 40.00% accuracy using ML pipeline and around 50.00% accuracy when using DL pipeline for both all and the optimal subset of electrodes.

Table 5.6: Accuracies obtained when introducing a third class, the rest class. The classifiers used are EEGnet and RF, with all electrodes and with electrode subsets found optimal previously.

Subject	Pipeline	Electrodes	Accuracy	SD
S03	EEGnet	all	95.50%	4.26%
		5(C3), 14	91.50%	3.09%
	CSP + RF	all	88.75%	7.36%
		4, 5, 7, 8(Cz)	80.00%	8.05%
S09	EEGnet	all	77.83%	7.00%
		2, 13	76.67%	6.34%
	CSP + RF	all	54.17%	9.42%
		8(Cz), 13	59.16%	6.45%
S11	EEGnet	all	73.67%	5.45%
		4,6,11(C4)	67.50%	6.38%
	CSP + RF	all	47.92%	12.92%
		2, 6, 7, 8(Cz), 11(C4), 14	53.75%	8.44%
S13	EEGnet	all	50.33%	8.05%
		2, 3, 4, 14	54.50%	6.04%
	CSP + RF	all	40.83%	4.40%
		9, 10, 11(C4), 13, 14	40.00%	9.25%

The findings of this experiment reveal that the previous optimization of electrode selection may not be applicable when introducing a third class. Consequently, a new electrode selection process was undertaken to address this issue. The electrode selection was separated into two models: a flat model and a hierarchical model.

The flat model directly classifies the 3 distinct classes, namely hand, foot, and rest. On the other hand, the hierarchical model first classifies between rest and task and subsequently classifies between the two different MI tasks. The structure of the flat and hierarchical models are shown in figures 5.7 and 5.8, respectively.

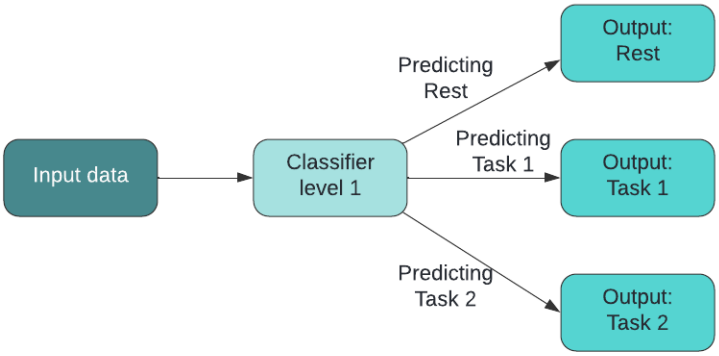


Figure 5.7: Flowcharts describing the steps of the flat classification model. Task 1 refers to the right-hand MI, and task 2 refers to feet MI

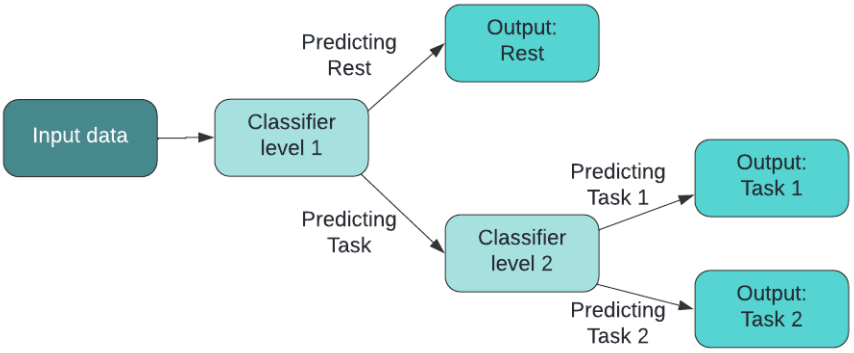


Figure 5.8: Flowcharts describing the steps of the hierarchical classification model. Task 1 refers to the right-hand MI, and task 2 refers to feet MI

5.5.1 Electrode Selection for Three Classes

Electrode Selection for a Flat Model

Due to the fact that electrode selection is a very time-consuming operation, it was only conducted for subjects S03 and S09 in the following sections. The selection of subjects S03 and S09 was done based on the good performance obtained in the previous sections. They are thus interesting subjects to experiment on when classifying 3 classes, namely right hand, feet, and rest MI.

Table 5.7 shows the results obtained from the electrode selection process for the 3-class classification. The table clearly shows that, for both subjects and pipelines, the new electrode selection resulted in an increase in accuracy. Further, the results implied that the EEGnet was the better classifier for 3 classes, corresponding to 2-class classification. Subject S03 obtained an accuracy of 100.00% with the EEGnet classifier and an accuracy of 90.00% for the CSP with RF, using the new optimal subset of electrodes. The usage of optimal electrodes led to an increase of 4.50% and 1.25% in accuracy for the EEGnet and RF, respectively. Similarly, subject S09 experienced accuracy improvements of 9.67% for EEGnet and 5.85% for CSP + RF, by using the newly selected optimal electrodes. Despite the increase in accuracy, subject S09 performed with lower accuracy compared to subject S03, particularly with the RF classifier. Regardless, both subjects got acceptable results, characterized by relatively high accuracies, especially when using the EEGnet classifier.

Compared to the previous electrode selection, using the NSGA to find the electrode subset yielding the highest accuracies, resulted in an increased number of electrodes for both subjects and pipelines when the third class was introduced. Furthermore, the majority of electrodes were selected from the left and middle parts of the head, aligning with the trend observed in the electrode selection for 2 classes on these subjects. For both subjects, the highest accuracies obtained were achieved with several combinations of electrodes, with minimal variability. The selected electrodes remained quite unchanged for a given accuracy and pipeline, with only minor substitutions involving their closest neighboring electrodes within the same region. Table 5.7 presents one of the electrode combinations which was used to obtain the highest accuracy.

Table 5.7: Accuracies obtained with NSGA, with optimized electrodes for a flat model with three classes. The three classes are right-hand, foot, and rest.

Subject	Pipeline	Electrodes	Accuracy
S03	EEGnet	6, 9, 14	100.00%
	CSP + RF	2, 4, 5(C3), 7, 8(Cz), 14	90.00%
S09	EEGnet	2, 6, 12, 13	87.50%
	CSP + RF	1, 4, 5(C3), 8(Cz), 13	65.00%

Electrode Selection for a Hierarchical Model

In the hierarchical model, the classification process is dived into two levels: classifier level 1 and classifier level 2. At classifier level 2, the first electrode selection that was conducted in the previous section remains applicable, as it focuses on the classification of the two MI tasks, namely right hand, and feet. However, to classify the rest and task states at classifier level 1, a separate electrode selection was conducted. The results of this electrode selection are shown in table 5.8.

Upon comparing table 5.8 with table 5.7 for the flat model, a notable difference in the electrode selection can be observed. Specifically, an increase in the number of electrodes needed to obtain the highest accuracy was observed for all pipelines in the hierarchical model, except for subject S03 using CSP + RF. Consequently, the number of electrodes located in the right section of the head increased. Notably, subject S03, which previously did not feature any electrodes from the right section for achieving the highest accuracy, now included one electrode from this region in the current electrode selection.

Furthermore, similar to previous electrode selection, the obtained highest accuracies were associated with multiple subsets of electrodes with some small differences. Table 5.8 shows one combination of the electrodes that resulted in the highest obtained accuracy for each pipeline. Another interesting observation was the consistently high accuracies attained across the subjects and pipelines. As both the level 2 classifier and the level 1 classifier obtained high accuracies, there was a hope that the hierarchical model could yield better results compared to the flat model.

Table 5.8: Accuracies obtained with optimized electrodes for classifier level 1 in a hierarchical model. Separating between rest and task.

Subject	Pipeline	Electrodes	Accuracy
S03	EEGnet	6, 8(Cz), 13, 14	96.90%
	CSP + RF	2, 3, 4, 5(C3), 7, 8(Cz)	90.63%
S09	EEGnet	2, 3, 4, 6, 8(Cz), 10, 13, 15	90.60%
	CSP + RF	2, 4, 7, 8(Cz), 10, 12, 13, 14, 15	79.06%

In the hierarchical model, when the two classifiers were merged into a single pipeline, the electrode selections made for the two classifier levels were included. Additionally, the hierarchical model was evaluated using all available electrodes. The recall of the various electrodes selected for the electrode subsets is listed below, while the corresponding results are presented in table 5.9.

- S03-EEGnet
  - Electrodes classifier level 1: 6, 8 (Cz), 13, 14
  - Electrodes Classifier level 2: 5 (C3), 14
- S03-CSP-RF
  - Electrodes classifier level 1: 2, 3, 4, 5 (C3), 7, 8(Cz)
  - Electrodes Classifier level 2: 4, 5 (C3), 7, 8

- S09-EEGnet
  - Electrodes classifier level 1: 2, 3, 4, 6, 8 (Cz), 10, 13, 15
  - Electrodes Classifier level 2: 2, 13
- S09-CSP-RF
  - Electrodes classifier level 1: 2, 4, 7, 8 (Cz), 10, 12, 13, 14, 15
  - Electrodes Classifier level 2: 8, 13

Table 5.9: Accuracies obtained from a three-class classification using a hierarchical model, separating between rest and two different MI-tasks.

Subject	Pipeline	Electrodes	Accuracy	SD
S03	EEGnet	all	95.13%	4.24%
		subset	86.88%	3.41%
	CSP+RF	all	87.19%	4.93%
		subsets	78.44%	6.77%
S09	EEGnet	all	77.63%	5.93%
		subsets	70.63%	4.19%
	CSP+RF	all	51.56%	11.03%
		subsets	66.88%	4.46%

An interesting observation from this experiment is that, except for the ML pipeline applied to subject S09, all pipelines obtained a drop in accuracy when utilizing the subset of optimal electrodes conducted from the two electrode selections. Moreover, the accuracies achieved by the hierarchical model are lower compared to the flat model, indicating that using a flat model is better for classifying 3 classes where two classes are MI and one class is rest. Further, subject S09 obtained random guessing when using the ML pipeline with all electrodes, although the SD, which was quite high, indicates a considerable variation in performance across multiple runs. Subject S03 using EEGnet gave the highest accuracies, as for every previous experiment. It is though worth noting that the obtained accuracies for subject S03 are lower than those obtained in the multiple other conducted experiments.

5.6 Classification of Motor Imagery using Transfer Learning

5.6.1 Transfer Learning using Deep Learning

TL is an up-and-coming approach for classification where knowledge gained from one domain is transferred to improve performance in another domain. In the context of EEG classification, this consists of training a model on a set of subjects and then updating the weights and biases to enable accurate predictions on another unseen subject. This way, there is a hope that bad-performing subjects can enhance their performance.

Subjects S03 and S09 have consistently shown good results during earlier experiments, whereas subject S13 has shown the ability to improve, but still being the lowest-performing subject. In addition, all three subjects have shown that their highest accuracies were obtained using a high percentage of electrodes on the left and middle section of the brain when an electrode selection has been conducted. Therefore, there is a potential opportunity to enhance the performance of subject S13 by employing a TL model trained on data from subjects S03 and S09.

In all the TL-experiments performed, a bandpass filter of $8-30Hz$ (mu-beta band) was applied to the data. The first TL experiment involved creating four different models, each employing a different subset of electrodes. The first model used all the electrodes, while the remaining 3 models used the 2, 3, and 4 best electrodes from electrode selection. The specific electrodes used for each model are as follows:

- 2 electrodes
  - S03: 5 (C3) and 14
  - S09: 2 and 13
  - S13: 2, 4
- 3 Electrodes
  - S03: 2, 6, and 14
  - S09: 2, 4, and 5 (C3)
  - S13: 2, 4, and 9
- 4 Electrodes
  - S03: 6, 8 (Cz), 9, and 14
  - S09: 2, 4, 6, and 9
  - S13: 2, 3, 4, 14

The TNR, TPR, and accuracy were recorded to get an indication of which TL model performed highly. Table 5.10 presents the scores obtained from the TL-models for the given electrode subsets. The table includes both the initial scores of the models without any data from subject S13 being incorporated and the classification scores of subject S13 after the model is fine-tuned to this subject. In addition, the SD of the model's performance when classifying S13 is provided.

Table 5.10: Accuracies, TPRs, TNRs, and SDs for TL models on subject S13, using different electrode subsets. "Model - without S13" shows the score obtained on the model when building it using training, validation, and a test set. "TL on S13" shows the score when the model was tuned on some of the data from S13 and then tested on the remaining data.

Electrode selection	Metrics	Model - without S13	TL on S13	SD
All Electrodes	TPR	100.00%	45.00%	13.60%
	TNR	90.00%	69.50%	6.50%
	Accuracy	95.00%	57.25%	5.18%
2 Electrodes	TPR	100.00%	73.00%	8.12%
	TNR	80.00%	62.50%	4.03%
	Accuracy	90.00%	**67.75**%	3.54%
3 Electrodes	TPR	100.00%	35.00%	0.00%
	TNR	95.00%	77.50%	4.03%
	Accuracy	97.50%	56.25%	2.02%
4 Electrodes	TPR	100.00%	65.00%	3.16%
	TNR	95.00%	62.50%	5.59%
	Accuracy	97.50%	63.75%	1.68%

Based on the results presented in table 5.10, subject S13 showed a slight improvement in accuracy when using all electrodes and the mu-beta band, increasing from 50.25% obtained in the initial tests to 57.25%. However, this accuracy is still insufficient to ensure whether it was not obtained by chance. This is supported by the SD value, showing that it is possible for this model to produce random guessing values. Similarly, using 3 electrodes yielded comparable results, with an accuracy that is not significantly higher than what can be obtained by chance.

In contrast, using both 2 and 4 electrodes showed more promising results. Both obtained average accuracies above 60.00% with low SD scores. Nevertheless, these accuracies were still lower than what was obtained by subject S13 alone during electrode selection, specifically 68.80% for 2 electrodes and 75.00% for 4 electrodes. Thus, further experiments were conducted to explore alternative TL models that could potentially yield an increase in the results.

The next experiment was done using the subset of 2 electrodes, considering that the accuracy achieved with this subset was not significantly different from the accuracy obtained from the electrode selection. Since subject S11 was the only other participant who underwent electrode selection, a model consisting of subjects S03, S09, and S11 was constructed to see if it could improve the results obtained in the first TL experiment. From the electrode selection, it was shown that using electrodes 10 and 13 gave the highest accuracy with 2 electrodes for subject S11, thus they were used in this experiment. The results obtained are shown in table 5.11.

Table 5.11: Accuracies, TPRs, TNRs, and SDs for TL model using the subset of 2 electrodes both for the model before the transfer and after tuned and tested on subject S13.

Metrics	Model - without S13	TL on S13	SD
TPR	100.00%	51.00%	4.90%
TNR	79.67%	74.50%	1.50%
Accuracy	88.33%	62.75%	2.36%

As seen from the table 5.11, including subject S11 into the model led to a decrease in performance by 5.00% compared to the model constructed only with subjects S03 and S09. Thus, continuing on models made from these 3 subjects was stopped. Further, it was tested whether using a higher number of subjects would result in improved models compared to a smaller subset of subjects. Given that only 4 out of the initial 14 subjects had undergone electrode selection, these experiments were conducted using all the electrodes. Three different models were made in this part of the experiment. The first model was a "leave-one-out" model, where all subjects except S13 were used to make the model. The next model consisted of only the subjects obtaining accuracies above 65% in the initial test. Lastly, a model consisting of all subjects obtaining a score under 65% (except S13) in the initial test was made. The results obtained from this experiment are shown in table 5.12.

Table 5.12: Accuracies, TPRs, TNRs, and SDs for TL models using different subsets of subjects, both for the model before the transfer and after tuned and tested on subject S13.

Subset selection	Metrics	Model - without S13	TL on S13	SD
Leave-one-out	TPR	86.92%	81.00%	4.36%
	TNR	65.38%	49.50%	1.50%
	Accuracy	76.15%	65.25%	1.75%
All above 65%	TPR	82.86%	75.00%	8.26%
	TNR	81.43%	57.72%	8.62%
	Accuracy	82.14%	66.36%	1.24%
All bellow 65%	TPR	64.23%	68.5%	2.29%
	TNR	61.43%	66.00%	4.36%
	Accuracy	62.86%	67.50%	1.75%

As observed in the table 5.12, the highest score was obtained when employing the model composed of all subjects whose initial scores were below 65%. However, it was still slightly lower than the result obtained from the model consisting of 2 electrodes made from subjects S03 and S09. Furthermore, using the "leave-one-out" approach and all subjects above 65% did not result in a significant decrease in accuracy compared to the best result obtained for subject S13 in the TL-experiments. These findings suggest that the inclusion of additional subjects, may not necessarily lead to improved performance in the TL models.

Furthermore, to investigate the applicability of TL for good-performing subjects, TL models were tested on a subject with higher initial accuracy, namely subject S09. As shown, the highest accuracy obtained so far by S09 was 93.80% by using electrode selection. Similar to subject S13, 3 models were constructed: the "leave-one-out" model, the "all-above-65%" model, and the "all-under-65%" model, as previously described. The results of these experiments are presented in table 5.13.

Table 5.13: Accuracies, TPRs, TNRs, and SDs for TL models using different subsets of subjects, both for the model before the transfer and after tuned and tested on subject S09.

Subset selection	Metrics	Model - without S09	TL on S09	SD
Leave-one-out	TPR	92.31%	99.17%	1.86%
	TNR	52.31%	90.83%	9.75%
	Accuracy	72.31%	95.00%	4.56%
All above 65%	TPR	83.33%	98.57%	2.26%
	TNR	83.33%	90.50%	8.10%
	Accuracy	83.33%	94.54%	3.30%
All bellow 65%	TPR	70.00%	78.47%	8.86%
	TNR	56.25%	93.33%	3.73%
	Accuracy	63.13%	91.25%	3.15%

As indicated in the table, subject S09 obtained high accuracies using the TL models. Both the leave-one-out and all-above-65% obtained higher accuracies than the initial test using all electrodes. In addition, these two models yielded higher results compared to the electrode selection approach. When employing the leave-one-out model, subject S09 achieved accuracies from 87.50% to 100.00%. These results are very good, especially the accuracy of 100.00% which has not yet been achieved by this subject. In addition, these results are highly favorable in comparison to earlier experiments, where numerous and computationally expensive steps were conducted, resulting in similar or lower performance.

The results presented in table 5.12 and table 5.13 show that it is possible to make good TL models using the leave-one-out method. Thus, this method was tested for all the subjects to see if the trend would yield for them all. The results are shown in table 5.14.

Table 5.14: Accuracies, TPRs, TNRs and SDs for Leave-one-out TL models for all the subjects in the dataset, except S09 and S13

Subject	Metrics	TL-Model	TL on Subject	SD
S01	TPR	67.69%	62.50%	36.46%
	TNR	84.62%	41.50%	36.90%
	Accuracy	**76.15%**	**52.00%**	**4.97%**
S02	TPR	97.69%	98.50%	2.42%
	TNR	45.38%	41.50%	10.81%
	Accuracy	**71.54%**	**70.00%**	**5.27%**
S03	TPR	85.38%	92.50%	5.89%
	TNR	60.77%	83.00%	5.87%
	Accuracy	**73.08%**	**87.75%**	**1.84%**
S04	TPR	98.46%	83.00%	8.56%
	TNR	32.31%	56.50%	30.10%
	Accuracy	**65.38%**	**69.75%**	**12.55%**
S05	TPR	75.38%	85.50%	6.85%
	TNR	75.38%	41.00%	11.50%
	Accuracy	**75.38%**	**63.25%**	**4.87%**
S06	TPR	98.56%	86.00%	11.50%
	TNR	46.92%	59.50%	22.30%
	Accuracy	**72.69%**	**72.75%**	**7.02%**
S07	TPR	83.85%	98.50%	2.42%
	TNR	63.08%	63.00%	10.33%
	Accuracy	**73.45%**	**80.75%**	**5.53%**
S08	TPR	73.08%	49.50%	12.79%
	TNR	74.62%	73.00%	10.85%
	Accuracy	**73.85%**	**61.25%**	**9.30%**
S10	TPR	86.92%	88.50%	10.81%
	TNR	56.92%	47.50%	7.98%
	Accuracy	**71.92%**	**68.80%**	**7.98%**
S11	TPR	85.38%	90.50%	7.98%
	TNR	63.85%	60.50%	20.47%
	Accuracy	**74.62%**	**75.50%**	**7.80%**
S12	TPR	92.31%	82.00%	14.18%
	TNR	59.23%	51.00%	14.87%
	Accuracy	**75.77%**	**66.5%**	**6.26%**
S14	TPR	75.38%	67.5%	12.96%
	TNR	76.15%	55.50%	14.23%
	Accuracy	**75.77%**	**61.50%**	**3.94%**

The results presented in table 5.14 reveal a mixed pattern for the accuracy obtained from the leave-one-out method. Approximately half of the subjects decreased their results compared to the initial test using the EEGnet classifier. Interestingly, the subjects that demonstrated an improvement in accuracy were primarily those who achieved close to random guessing in the initial test, now obtaining an accuracy above 60.00%. This seemed to be the trend for most of the subjects, except for S01, which stayed at random guessing even after the transfer, and S09 which was shown to get good results from the TL-models.

These observations suggest that the effectiveness of the leave-one-out method for enhancing classification accuracy varies among subjects. While it proved beneficial for subjects with initially poor performance, it did not consistently lead to improvements for all individuals.

The SD associated with each subject was on average quite high, particularly for subject S01. The high SD values indicate that the accuracies obtained from the 10 runs conducted for each subject yielded significant variations in the accuracy. This variability suggests that different runs of the TL models can yield both higher and lower accuracies than those obtained in the initial tests. Thus, which data the model uses to update the weights and biases seems to be important for the accuracy of the model.

5.6.2 Transfer Learning using Machine Learning

Subject-independent models using the leave-one-out ML approach can be seen as a form of TL model. Therefore, we also investigated whether making these types of models could yield promising results compared to DL models. To investigate this, subjects S09 and S13 were evaluated using the same experimental setup as for DL, specifically the band-pass filtering of the data between $8-30Hz$. Further, the pipeline used to make the models were CSP with 7 features and RF as this was shown to give high accuracies for all the subjects in the pipeline selection.

First, models for subject S13 were made using subsets of electrodes. As seen from the electrode selection, a higher number of electrodes was needed to obtain high accuracies for the ML models compared to the DL models. Specifically, subject S13 achieved the highest and second-highest accuracy when using 9 electrodes, indicating the importance of including a higher number of electrodes in the TL models. Further, using 5 electrodes resulted in only a small decrease in accuracy compared to the 9-electrode subset. Thus, the 3 models made for TL are composed of all electrodes, 9 electrodes, and 5 electrodes.

The electrodes associated with the highest accuracies for both the 5-electrode and 9-electrode subsets are provided below. The corresponding results obtained from the TL models corresponding to these electrode configurations are presented in table 5.15:

- 5 electrodes
  - S03: 3, 4, 5 (C3), 6, 8 (Cz)
  - S09: 5 (C3), 8 (Cz), 13, 14, 15
  - S13: 9, 10, 11 (C4), 13, 14
- 9 electrodes
  - S03: 1, 3, 4, 5 (C3), 6, 7, 8 (Cz), 10, 14
  - S09: 1, 2, 4, 7, 9, 12, 13, 14, 15
  - S13: 2, 3, 4, 8 (Cz), 9, 10, 12, 13, 15

Table 5.15: Accuracies, TPRs, TNRs, and SDs for TL models using different subsets of electrodes, both for the model before the transfer and when tested on subject S13.

Electrode selection	Metrics	Model - without S13	TL on S13	SD
All Electrodes	TPR	90.00%	7.50%	7.07%
	TNR	85.00%	92.25%	7.44%
	Accuracy	87.50%	51.88%	5.30%
5 Electrodes	TPR	95.00%	0.00%	0.00%
	TNR	90.00%	100.00%	0.00%
	Accuracy	92.50%	50.00%	0.00%
9 Electrodes	TPR	95.00%	100.00%	0.00%
	TNR	85.00%	0.00%	0.00%
	Accuracy	90.00%	50.00%	0.00%

The results presented in table 5.15 indicate that the subject-independent models constructed from subjects S03 and S09 only achieved random guessing when classifying data from subject S13. In addition, for both models made from the subset of electrodes, every individual data point from subject S13 got classified into the same class, resulting in 100.00% TPR or TNR, and 50.00% accuracy.

Since the results from the electrode subsets were far from what we hoped for, the leave-one-out method was tested for subject S13 to see if it could yield better results. As for DL-models, this consisted of one "leave-one-out", one "all-above-65%", and one "all-under-65%"-model. The results from this experiment are shown in table 5.16.

Table 5.16: Accuracies, TPRs, TNRs, and SDs for TL models using different subsets of subjects, both for the model before the transfer and when tested on subject S13.

Subset selection	Metrics	Model - without S13	TL on S13	SD
Leave-one-out	TPR	71.54%	18.75%	18.85%
	TNR	61.53%	88.75%	14.58%
	Accuracy	66.54%	53.75%	6.41%
All above 65%	TPR	64.29%	0.00%	0.00%
	TNR	80.00%	100.00%	0.00%
	Accuracy	72.14%	50.00%	0.00%
All bellow 65%	TPR	60.00%	27.50%	42.34%
	TNR	61.43%	63.75%	37.77%
	Accuracy	60.71%	45.63%	9.80%

From the table 5.16, it can be seen that using these models yields low accuracies similar to those obtained in the TL experiment using electrode subsets. Moreover, similar to the cases involving 5 and 9 electrodes, the "all-above.65%" model yielded a classification of every data point into one class. In addition, the SDs from this experiment were relatively high.

In the final experiment, an assessment was made to determine if subjects who achieved higher initial scores could benefit from TL. For this experiment, the same tests comprising different leave-one-out models were conducted for subject S09, with the results presented in table 5.17.

Table 5.17: Accuracies, TPRs, TNRs, and SDs for TL models using different subsets of subjects, both for the model before the transfer and when tested on subject S09.

Subset selection	Metrics	Model - without S09	TL on S09	SD
Leave-one-out	TPR	71.54%	43.75%	27.74%
	TNR	53.08%	83.75%	20.66%
	Accuracy	62.31%	63.75%	10.61%
All above 65%	TPR	73.33%	41.25%	23.57%
	TNR	75.00%	87.50%	19.09%
	Accuracy	74.17%	64.38%	8.21%
All bellow 65%	TPR	60.00%	51.25%	21.00%
	TNR	43.75%	55.00%	22.68%
	Accuracy	51.88%	53.13%	12.23%

Similar to subject S13, subject S09 also got low results when subjected to the different TL models. Compared to the initial test, the accuracies were between 13.37% to 30.68% lower, while compared to TL using the EEGnet they were between 30.16% to 38.12% lower. This shows that subject-independent models work worse than subject-dependent models, especially after electrode selection is done for the subject-dependent models. In addition, these results show that TL using ML yields worse results than TL using DL.

5.7 Online Classification with Drone Actuation

5.7.1 Time Widow Optimization

When an online classification is done, the data segmentation differs slightly from that employed in offline classification. In experiments conducted prior to this section, the time period was set to 3 seconds. However, to enable the possibility of performing live simulations, we wanted to reduce the time period to 1 second, yielding new classifications at one-second intervals. In order to find the best time window to use in these models, a time window optimization was performed. To reduce the time used in this section, the number of subjects was decreased to only subjects S03 and S09.

Since the online classification was performed for both 2 and 3 classes, the time window optimization was conducted accordingly. The resulting tables are shown below, where table 5.18 shows time optimization results for 2 classes, and table 5.19 shows time optimization results for 3 classes. In both tables, the time window columns indicate the specific second yielding the highest accuracy, where 0 seconds denotes the beep moment, as shown in figure 4.3.

Table 5.18: Time window optimization for 2 classes on subject S03 and S09. Where the time window indicates which second gives the best accuracy from the beep. The optimization is done for both all electrodes and for the optimal electrodes found in the electrode selection.

Subject	Pipeline	Electrodes	Time Window	Accuracy
S03	EEGnet	all	4.50s - 5.50s	100.00%
		5(C3), 14	3.50s - 4.50s	97.50%
	CSP+RF	all	3.50s - 4.50s	99.38%
		4, 5(C3), 7, 8(Cz)	3.50s - 4.50s	98.75%
S09	EEGnet	all	5.25s - 6.25s	97.50%
		2, 13	7.00s - 8.00s	97.50%
	CSP+RF	all	5.50s - 6.50s	95.00%
		8(Cz), 13	6.25s - 7.25s	95.00%

From table 5.18 it can be seen that S03 and S09 got their highest accuracy scores in two completely different time windows, where S03 has its highest scores earlier than S09. Furthermore, using the optimal electrodes resulted in a slight decrease in accuracy for S03, whereas it remained consistent for S09. In addition, for S09 it was also a bit bigger gap between which time window lead to the highest accuracies when all vs optimal electrodes were used.

Table 5.19: Time window optimization for 3 classes on subjects S03 and S09. Where the time window indicates which second gives the best accuracy from the beep. The optimization is done for both all electrodes and for the optimal electrodes found in the electrode selection.

Subject	Pipeline	Electrodes	Time Window	Accuracy
S03	EEGnet	all	3.25s-4.25s	96.67%
		6, 9, 14	4.00s - 5.00s	96.67%
	CSP+RF	all	3.25s - 4.25s	88.75%
		2, 4, 5(C3), 7, 8(Cz),14	3.50s - 4.50s	90.42%
S09	EEGnet	all	4.50s - 5.50s	86.67%
		2, 6, 12, 13	4.50s - 5.50s	81.67%
	CSP+RF	all	4.00s - 5.00s	77.92%
		1, 4, 5(C3), 8(Cz), 13	5.00s - 6.00s	77.08%

From table 5.19 it can be seen that the time window used to obtain the highest accuracies for the two different subjects was more similar compared to the 2 classes. Specifically, both of the two subjects obtained their highest accuracies between 3-5 seconds after the beep. Consequently, subject S03 used approximately the same time window to obtain high accuracies as for the 2 classes, while the time window for S09 shifted to some seconds earlier. Further, as seen from the results, subject S03 achieved higher accuracies using the ML pipeline and the optimal electrodes compared to not using the optimal electrodes, while remaining unchanged using the DL pipeline. On the other hand, subject S09 experienced a small decrease in accuracies for both pipelines using the optimal electrodes compared to using all electrodes.

5.7.2 Actuating Drone using 2 Classes

To facilitate the streaming of 2 MI tasks for classification purposes, the rest state data was separated from the data, resulting in a stream exclusively containing MI tasks. Specifically, the task stream consisted of the time period between 2 and 8 seconds after the beep, as it was shown in the time window optimization that the data before and after these points never gave a satisfactory classification of MI-tasks. Further, the task stream was segmented into 1-second intervals, giving a new classification each second.

Since the drone moves for 1.5 seconds when classifying foot, it ignores the new commands in this time period avoiding making a queue of commands. Additionally, all electrodes were used as they most often showed the highest or comparable accuracy compared to the optimal electrodes for the time window-optimized models. To see how well the model performed, the accuracy was recorded. To identify the time windows associated with the most accurate results, the corresponding time windows along with their associated accuracies were also recorded. For subject S03, these results are presented in tables 5.20 and 5.21.

Table 5.20: Accuracy, TPR, and TNR obtained when actuating a drone with 2 classes, right-hand and foot MI, for subject S03. The time window lasts from 2 to 8 seconds following the beep and is segmented into 1-second classification windows.

Classifier	Metrics	Accuracy
EEGnet	TPR	83.33%
	TNR	80.00%
	Accuracy	81.67%
CSP+RF	TPR	88.33%
	TNR	90.00%
	Accuracy	89.17%

Table 5.21: Accuracies obtained in the different time windows when classifying 2 MI-tasks, right-hand and foot, for subject S03.

Pipeline	Time Window	Accuracy
EEGnet	2s-3s	30.00%
	3s-4s	100.00%
	4s-5s	100.00%
	5s-6s	95.00%
	6s-7s	85.00%
	7s-8s	80.00%
CSP + RF	2s-3s	60.00%
	3s-4s	100.00%
	4s-5s	100.00%
	5s-6s	95.00%
	6s-7s	80.00%
	7s-8s	100.00%

As seen from the results in table 5.20, the accuracy decreased significantly from the time window-optimized models. Still, the accuracies are quite good, with CSP+RF giving higher classification accuracy than the EEGnet. In addition, the TPR and TNR were quite even, indicating that they classified the hand and foot MI to the same extent.

Based on the results presented in table 5.21, it can be seen that the highest classification performance was observed within the time window spanning from 3 seconds to 6 seconds, in addition to the second 7 to 8 for RF. During these intervals, both pipelines had accuracies ranging from 95.00% to 100.00%. In addition, the remaining seconds yielded lower accuracies, thus being the reason for the drop in accuracy for EEGnet in table 5.20.

The same experimental procedure was performed for subject S09, leading to the results shown in table 5.22 and table 5.23. As seen from these results, EEGnet classified better than RF in contrast to subject S03. Remarkably, it achieved a higher accuracy score than S03 when using the EEGnet, which was the first time subject S09 outperformed subject S03 in our experiments. Further, the results show that the classification was to some extent biased towards the foot task, leading to a high TNR and a low TPR.

Table 5.22: Accuracy, TPR, and TNR obtained when actuating a drone with 2 classes, right-hand and foot MI, for subject S09. The time window lasts from 2 to 9 seconds following the beep and is segmented into 1-second classification windows.

Classifier	Metrics	Accuracy
EEGnet	TPR	71.67%
	TNR	96.67%
	Accuracy	84.17%
CSP+RF	TPR	66.67%
	TNR	98.33%
	Accuracy	82.50%

Table 5.23 reveals that the EEGnet achieves the highest classification scores within the time window spanning from 4 seconds to 7 seconds, while the RF classifier attains the highest scores within the interval of 4 seconds to 8 seconds. In these intervals, both models achieved high classification scores, ranging from 95.00% to 100.00%. On the other hand, the accuracy scores were significantly lower in some of the remaining seconds.

Table 5.23: Accuraies obtained in the different time windows when classifying 2 MI-tasks, right-hand and foot, for subject S09.

Pipeline	Time Window	Accuracy
EEGnet	2s-3s	55.00%
	3s-4s	60.00%
	4s-5s	95.00%
	5s-6s	100.00%
	6s-7s	100.00%
	7s-8s	95.00%
CSP + RF	2s-3s	55.00%
	3s-4s	55.00%
	4s-5s	95.00%
	5s-6s	100.00%
	6s-7s	95.00%
	7s-8s	95.00%

5.7.3 Actuating Drone using 3 Classes

Similar to the section on actuating a drone using 2 classes, this experiment used all the electrodes and segments the data into 1-second intervals for classification. Furthermore, the same time period was marked as task data similar to the previous section, while the remaining data was marked as rest. The models were constructed using the time-optimized window found in the corresponding experiment. The resulting accuracies obtained from this experiment are shown in tables 5.24 and 5.25.

Table 5.24: Accuracy obtained when actuating a drone using 3 classes, right-hand, foot, and rest, where the model is trained on the time optimized window. The task time lasts from 2 to 8 seconds following the beep, the remaining data is marked as rest.

Subject	Pipeline	Accuracy
S03	EEGnet	68.18%
	CSP+RF	65.00%
S09	EEGnet	61.19%
	CSP+RF	67.12%

Table 5.25: Accuracies obtained for the individual 3 classes, right-hand, foot, and rest, where the model is trained on the time optimized window. The task time lasts from 2 to 8 seconds following the beep, the remaining data is marked as rest.

Subject	Pipeline	Time Window	Accuracy
S03	EEGnet	Hand MI	88.33%
		Foot MI	71.67%
		Rest	54.00%
	CSP-RF	Hand MI	81.67%
		Foot MI	75.00%
		Rest	49.00%
S09	EEGnet	Hand MI	75.00%
		Foot MI	91.67%
		Rest	34.34%
	CSP+RF	Hand MI	60.00%
		Foot MI	85.0%
		Rest	60.61%

The results from classifying 3 classes show that the accuracy decreases significantly compared to the 2-class classification. Moreover, the classification accuracy of the rest state was on average significantly lower for both subjects and classification methods compared to the right-hand and foot MI task, except for subject S09 using RF. Thus, the classification of the rest state was often the leading factor for the low accuracies obtained for all the 3 classes. Further, for S03 the classification of right hand MI gave higher accuracies than foot MI, while the opposite was found for S09. It was also found that subject S03 gave higher scores than subject S09 using EEGnet, while subject S09 obtained higher score than subject S03 using RF. As the classification of rest data was on average quite low, a new strategy was purposed for 3 class classifications. Instead of only relying on the previously determined 1-second time window, found from time window optimization, for the training of the classifiers, the model was made using data from every second within the recordings. Specifically, seconds 2 to 8 following the beep were marked as MI-task, while the remaining was marked as rest. The results are presented in tables 5.26 and 5.27.

Table 5.26: Accuracies obtained flying a drone using 3 classes, right-hand MI, foot MI, and rest, where the model is trained on the whole data set. Task being between 2 to 8 sec following the beep, the remaining data is marked as rest.

Subject	Pipeline	Accuracy
S03	EEGnet	77.27%
	CSP+RF	72.27%
S09	EEGnet	74.89%
	CSP+RF	64.38%

Table 5.27: Accuracies of the individual 3 classes right-hand MI, foot MI and rest, where the model is trained on the whole data set. Task being between 2 to 8 seconds following the beep, the remaining data is marked as rest.

Subject	Pipeline	Time Window	Accuracy
S03	EEGnet	Hand MI	83.33%
		Foot MI	63.33%
		Rest	82.00%
	CSP-RF	Hand MI	78.33%
		Foot MI	70.00%
		Rest	70.00%
S09	EEGnet	Hand MI	75.00%
		Foot MI	88.33%
		Rest	66.67%
	CSP+RF	Hand MI	55.00%
		Foot MI	56.67%
		Rest	74.74%

These results obtained demonstrated that using all data from the subjects enhanced the accuracies compared to using the time-optimized model. The classification accuracy increased for subject S03 by 9.09% and 7.27% for EEGnet and RF respectively, while it increased by 13.70% for subject S09 using EEGnet. On the other hand, there was a small decrease of 2.80% for subject S09 using RF. Table 5.27 also shows that the classification of the rest state significantly improves compared to the results obtained from the time-optimized window. On the other hand, the classification of the MI tasks decreased to some extent. Additionally, the accuracies were still lower compared to the 2-class classification, as expected.

5.7.4 Video of Drone Actuations

The video linked in figure 5.9 shows two test runs of the actuation of the drone, one for 2 MI classes (right hand and foot) and one for 3 classes (right hand, foot, and rest). Both of the test runs use only a selection of all the data in the 2 different streams, thus resulting in a bit different accuracies than those obtained in the previous sections using the whole stream. In addition, both videos use data from subject S03. The video can also be reached following this link: https://youtu.be/SVvtyDZMIqg

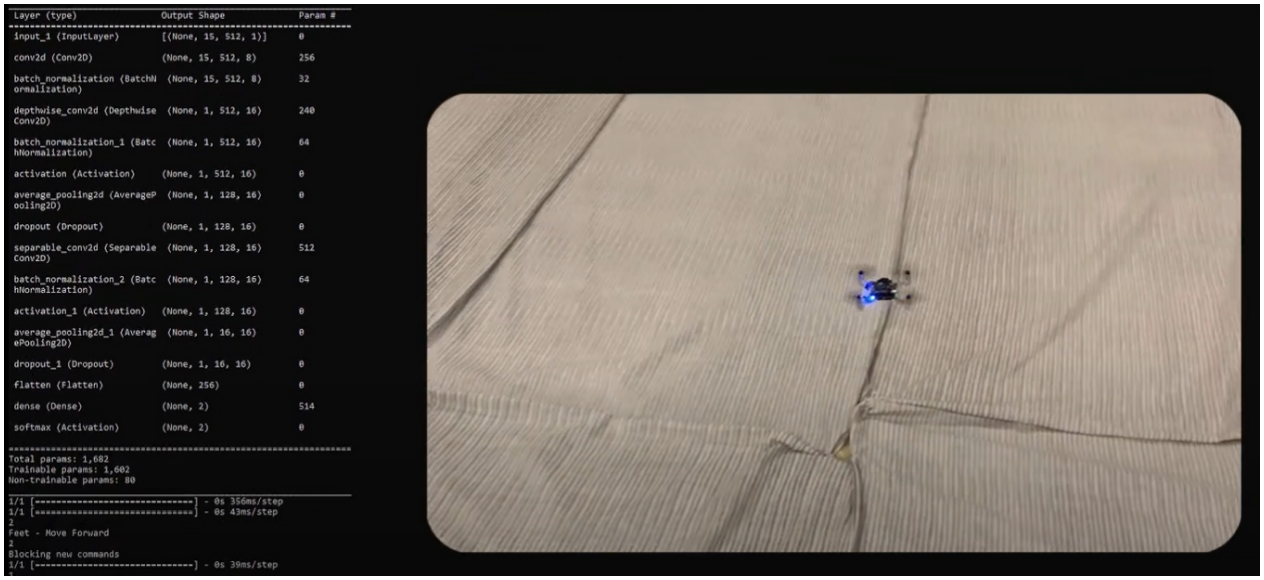


Figure 5.9: Hyperlink to the video of drone actuation using both 2 and 3 classes.

Chapter 6

Discussion, Conclusion and Further Work

6.1 Discussion

6.1.1 Feature Selection

An examination of the heatmaps presented in figures 5.2, 5.3, 5.4, and 5.5, reveals that subjects s03, s09, and S11 have LL as one of the four features occurring often when high accuracies were obtained. This fact suggests that this specific feature may in general be a feature that captures the characteristics of MI-signals well. On the other hand, LL is one of the features occurring the least times for S13. When comparing the four subjects in general, we see that S13 is by far the worst performer. Thus, the absence of LL for S13 might be an indication that other features are needed to capture the essence of MI-signals for low performers.

A contradiction to the notion that some features are important for good performers while different features are important for bad performers is seen when comparing subjects S03 and S09. Both of them are both very good performers, but they need quite different combinations of features to obtain high accuracies. Furthermore, a general observation is that all four subjects exhibit different feature combinations to obtain high accuracies, where some subjects might have a feature among the top four, while others have the same feature among the bottom four. The randomness observed in the features highlights the fact that feature selection is highly subject-dependent and may even vary across different sessions. This makes it challenging to identify features that are effective for all subjects.

6.1.2 Pipeline Selection

From the results obtained in the pipeline selection, it becomes evident that the EEGnet classifier outperforms all the ML pipelines. The reason behind this might be that the EEGnet is specifically designed to capture the essence in EEG signals, whereas the ML pipelines are more general methods applicable to many different domains[70]. Additionally, within the ML pipelines, an almost consistent observation was that the RF classifier outperformed the other classifiers for all the subjects, especially when combined with features calculated using CSP. This finding correlates to what was found in the literature review, which was that the RF classifier could outperform more conventional classifiers such as SVM and LDA when classifying MI-tasks, especially when there is limited data.

Through the literature review, it was found that CSP has demonstrated good results when classifying MI-tasks, which aligns with our findings from the pipeline selection. On the other hand, the literature review also states that DWT is a powerful method as the bands can contain different information about the MI action. This is a contradiction to our findings, showing that using DWT often yielded the worst results out of the different feature extraction methods. Several factors could contribute to the better performance of CSP over DWT. Firstly, CSP is effective in reducing the impact of noise and artifacts on EEG signals. Secondly, another reason might be because we excluded some of the DWT bands after the initial test, which might have contained important information about the EEG signal.

Further, our results showed small differences in accuracies when using different numbers of CSP features. In theory, employing too few levels may fail to capture the complexity of a signal. Given the complexity of EEG signals, the reasoning is that a higher number of CSP features are needed to obtain high accuracies. This hypothesis was supported by the findings for subject S13, where the highest accuracy was achieved when utilizing 7 CSP-features in the ML pipeline. On the other hand, for subject S03, the range of 3 to 7 features sufficiently yielded the same accuracy. Nevertheless, it is worth highlighting that subject S03 is a high-performing subject, obtaining high accuracies despite the pipeline chosen.

The final point from the results worth attention is that to obtain high accuracies, the mu-beta band was often used. This corresponds to the theory saying that the MI signals often occur within the frequency band of $8Hz$ to $30Hz$, corresponding to the mu-beta band. On the other hand, subject S03 gets its best results on the ML pipeline only using the beta band, while subject S13 gets its best results on the DL pipeline using only the mu band. This might be an indication that MI-signals might occur a bit differently across subjects. Nevertheless, both bands are in the mu-beta band and the drop in accuracy was not substantial when using the whole mu-beta band instead of only one part of the band.

6.1.3 Electrode Selection

According to theoretical expectations, during the performance of right hand and foot MI tasks, signal activation should primarily occur in the left and middle regions of the brain. Analysis of the results from electrode selection reveals that when using few electrodes to obtain high accuracies, these two specific sections were mostly used for subjects S03, S09, and S13. On the other hand, it was observed that subject S11 achieved higher accuracies when using electrodes from the left and the right sections instead of the middle and left, which contradicts the theoretical expectations. This emphasizes that the MI-signals and collection of the signals are very dependent on the subject itself. Several factors might be the explanation for these observations. One of which is the shape of the subject's head, as different head shapes may result in slightly different placements of the electrodes for each subject, leading to deviations from the anticipated placement compared to the brain region locations. Another factor might be the physiological differences among subjects, making a variability in where the signal origins within the brain for the different subjects[71].

Another observation from electrode selection was that the ML pipelines needed a higher amount of electrodes in order to obtain high accuracies compared to the EEGnet. One possible explanation might be that NNs often are prone to overfitting when confronted with an excessive amount of training data, resulting in a reduction in test accuracy. Reducing the number of channels can thus serve as one way of overcoming this problem. Another explanation, as previously mentioned, the EEGnet architecture is specifically designed to capture the essence of EEG signals, while the ML pipelines are not. Therefore, the EEGnet might possess a greater capability to identify the essential features necessary to differentiate between the MI tasks, thereby needing a lower number of electrodes to obtain high accuracies.

Prior to the electrode selection process, a Laplacian rereferencing technique was applied to reduce the number of electrodes. The obtained results revealed a combination of minor and substantial decreases in accuracy compared to the scenario where all electrodes were utilized. In addition, when CAR was introduced after the electrode selection, the accuracies either reminded the same or decreased. A possible explanation for this decrease might be that rereferencing can change the nature of the recorded signal and alter the relationships between electrodes. Specifically, rereferencing can affect the amplitude and polarity of the recorded signals, which in turn can affect the analysis results. Another explanation might be that instead of removing noise, it can enhance it. This means that if a reference electrode is noisy, the noise can be introduced in the rereferenced signal, consequently compromising the overall quality of the data.

6.1.4 Classification of Three Classes

Upon introducing a third class, it was observed that the overall accuracies decreased compared to the binary classification problem. This decrease in accuracy can be considered reasonable and expected, as a multi-class classification model has more complex decision boundaries between the classes than a binary model.

The results obtained from the 3-class classification analysis showed that the flat model gave higher accuracies than the hierarchical model. This outcome might be attributed to the flat model's ability to capture complex interactions between the features of all the classes, which might not be represented in the hierarchical structure. In addition, the first step in the hierarchical model is differentiating between rest and task, where the task is both the right hand and the foot. As the task is a combination of two different classes, it might be harder to find any specific features to capture the essence of both, thus making it harder to do the classification.

As seen from the results, the flat model needs more electrodes to obtain its highest values when classifying 3 classes compared to 2 classes. In addition, for the hierarchical model, many electrodes are needed to differentiate between rest and MI-task. For the flat model, the reason is probably that it is more complex to classify 3 classes over 2 classes, thus more information is needed to obtain a good classification model. For the level 1 classifier in the hierarchical model, the reason might be that it is more complex to differentiate between rest and task, especially since the task class is a combination of two distinct classes. Thus, more electrodes are needed to capture the essence of the MI-task class.

A surprising result obtained from the hierarchical model was that the final 3-class classification accuracies were higher using all the electrodes than the optimal electrodes found from electrode selection. One reason might be that more data is needed to obtain good classifications when 3 classes are introduced, leading to low-performing classifiers using a subset of electrodes.

6.1.5 Transfer Learning

The results obtained from the TL-models showed that S13 gets its highest accuracies using a model composed of subjects S03 and S09 while employing only the 2 best electrodes selected from electrode selection. However, including S11 in the TL-model led to a decrease in accuracy. One possible explanation for this might be that subjects S03 and S09 have more similar EEG patterns to subject S13 than subject S11. Therefore, when introducing subject S11 to the model, it rather decreases the accuracies than increases it. This hypothesis is backed up by the model accuracy before the transfer is done, where it can be seen that it decreases by 27.25% when subject S11 is introduced compared to the model without subject S11. This is supported by studies highlighting that to produce good TL models, using subjects with similar EEG patterns are preferred[63].

The obtained results also indicate that subject S13 achieves higher accuracy when using 2 and 4 electrodes, as opposed to 3 electrodes. From the initial electrode selection on subject S13, it was found that the accuracy using 2 electrodes was 71.90%, using 3 electrodes was 68.80% while using 4 electrodes was 75.00%. Therefore, logical reasoning can be made that using 3 electrodes leads to lower results, potentially due to less precise updates of weights and biases. However, it is noteworthy that 2 electrodes outperformed 4 electrodes despite the latter initially demonstrating better results, which appears somewhat strange.

When subjects with initial test accuracies below 65.00% were employed, a modest improvement in performance for subject S13 was observed compared to using subjects with accuracies above 65.00%. Similarly, for the results when using subject S11 in the model, the same reasoning can be applied to these results. Specifically, the subjects with low initial accuracies have more similar EEG patterns than those with higher accuracies. On the other hand, subject S09 achieved its highest accuracy using the leave-one-out method. This means that despite using subjects with potentially dissimilar EEG patterns, the accuracy increases.

The only good-performing subject, increasing its accuracy after the leave-one-out TL-models was S09. At variance, the remaining subjects experienced a decrease between 0.75% to 16.75%. One plausible explanation for the decreases in accuracy among good-performing subjects is the potential negative impact of including low-performing subjects, making it harder to find common patterns and thus making it harder to predict the MI-task. In addition, a negative transfer can happen from a too-different or too-close-looking task, making the models decrease their performance after the transfer. However, the perplexing aspect lies in the fact that subject S09 increases its scores compared to the other good-performing subjects.

Further, the TL models utilizing DL outperformed the models made utilizing ML. The primary factor for this likely stems from the abilities of DL to update the weights and biases using some of the targeted subject data prior to the classification phase. In contrast, when performing TL using ML, there is no direct update of the model. Thus, the model must classify based only on what is learned from the subjects used for training.

A strange observation was found when using ML-TL models on subject S13. The results showed that using the all-above-65%, a subset of 5 electrodes, and a subset of 9 electrodes models led to a consequential classification of all data to the same class, making the accuracy 50% and either TPR or TNR 100.00%. It is worth mentioning that this phenomenon exclusively occurs for subject S13, which is generally considered a low-performing subject. Consequently, one possible reason for this behavior is that the other subjects have different EEG patterns from S13, making it hard to do a classification of this subject.

6.1.6 Online Classification with Drone Actuation

Through the process of the time window selection on two classes, it was observed that subjects S03 and S09 required two distinct time windows to obtain their respective peak accuracies. This results from the fact that subject S03 likely has a quicker performance of MI than subject S09. Therefore, this result indicated that the remaining subjects might increase their accuracies by applying the optimal time window specific to their MI occurrence.

Furthermore, the experiments showed that utilizing optimal electrodes resulted in a decrease in accuracy for both the 2-class classification on subject S03 and the 3-class classification on subject S09. However, it is important to note that the electrodes were optimized for a 3-second time window, rather than a 1-second window. Consequently, there might be other electrodes that are optimal for different lengths of time windows. Further, the time window shifted for S09 when classifying 3 classes compared to the classification of 2 classes. This shift suggests that the ability to distinguish the rest state from the tasks may be easier in an earlier time window than discriminating between the two tasks individually.

During the live simulation for 2 classes, the accuracies decreased noticeably compared to the window optimization results. The main reason for this can be seen in the tables 5.21 and 5.23, where it is shown that not all seconds marked as tasks yield good classification results. This discrepancy likely arises from the timing misalignment between the actual MI performance and the analysis window. Specifically, the subject had not started performing the MI or was finished performing it before the time was up.

The 3-class classification results using the time-optimized window show a decrease in the accuracies compared to the 2-class classification. This decline can be attributed to several factors. Firstly, classifying 3 classes is more challenging compared to classifying 2 classes. Further, the most noteworthy decrease in accuracy stems from the low classification of the rest state. The rest state might be classified wrongly because these seconds can be quite variable, having artifacts that are not seen in the 1-second used for model making. Furthermore, the classification of the two MI-tasks was also not exceptionally high. This outcome is probably the same as for 2 class classification described above.

Constructing the models utilizing all the data points within the training set resulted in improvements in accuracy, especially for the rest class. This outcome supports the hypothesis that the essence of the rest class may not have been fully captured when using the time-optimized window. Thus, it seems that using the time-optimized window might lack the ability to capture what is needed to classify more than 2 classes, especially the rest class. On the other hand, the accuracies for the 2 MI tasks decreased to some extent. This can be attributed due to the training of the model, where some data marked as task might have fitted better as rest, or opposite, making the classes less distinct.

The results also showed that the classification of the different classes was dependent on the subject. For subject S03, the classification accuracy of the right hand MI was higher than the feet, whereas the opposite was shown for subject S09. These results probably stem from the fact that subjects have different abilities to perform MI, especially the ability to perform some MI-tasks to a greater extent than others.

6.2 Conclusion

The primary objective of this thesis was to implement a system capable of actuating a drone based on MI signals. This was partly successfully carried out through the implementation of classification models distinguishing between two MI-task classes and two MI classes in addition to the rest. The accuracies from these experiments indicate the need for further research to improve the reliability of the models. Particularly, the aim is to minimize misclassification, ensuring that individuals, such as those with impairments, can utilize the model efficiently and safely. The finding in this study also showed that the classification of different MI-tasks was very subject-dependent, making it more complex to find a subject-independent model.

Prior to the live implementation, different pipelines were tested to identify the best models for different subjects. The steps in finding these models included feature selection, pipeline selection, and electrode selection using NSGA. The electrode selection was compared to Laplace rereferencing and CAR. In addition, TL was tested to see if transferring knowledge across subjects could enhance the model accuracies. From the results, it was seen that TL worked well for subject S09, but in general it did not lead to higher scores for most of the subjects. Thus, we found that using EEGnet with optimal electrodes yielded in general best accuracies for two-class classification.

Furthermore, the study revealed that the selection of manual features required to achieve high accuracies was very subject-dependent. Similarly, the optimal electrode selection for enhancing accuracy was dependent on the subject. However, a consistent trend was observed among 3 out of 4 tested subjects, where electrodes positioned on the left and middle sections were frequently used when a high accuracy score was obtained, which aligns with the theory.

The highest achieved results were obtained on subject S03, achieving a 100.00% accuracy score on offline classification. Subject S03 consistently performed well across every pipeline employed in this thesis, compared to other subjects where the results varied more. For instance, subject S11 displayed a wide range of accuracy scores, ranging from 43.75% in the initial test using the whole frequency band, all electrodes, and CSP+RF pipeline to 93.80% using EEGnet, the mu-beta band, and the optimal 3 electrodes. These findings underscore the effectiveness of the implemented strategies aimed to enhance accuracy. Furthermore, they show the importance of employing comprehensive steps, similar to the ones adopted in this study, to develop robust and high-performing models.

Table 6.1 shows the best-obtained results from the optimization part of this thesis compared to the results from the study done by David Steryl et al.[67] and the study done by Yang, Wang, and Huang[68]. Our results were obtained by exploiting the optimal electrodes from the electrode selection, which were as follows: subject S03 used 5(C3) and 14, subject S09 used 2 and 13, S11 used 4, 6, and 11(C4), while subject S13 used 2, 3, 4, and 14. Meanwhile, the results from David Steryl et al. were obtained using the Laplacian rereferencing scheme, while Yang, Wang, and Huang do not state if they use all 15 electrodes or the Laplacian rereferencing scheme. All three studies have used different methods for obtaining the results. Our thesis found EEGnet to be the most optimal, while David Steryl et al. obtained their highest accuracies using fbCSP-RF. Yang, Wang, and Huang obtained their best results using Euclidean space data alignment(EA) combined with CSP and CNN. Further, the results from table 6.1 show that our approach has given the highest accuracies for all subjects, except for subject S13. From this, it is clear that the approaches implemented in this thesis were successful and worth looking more into.

Table 6.1: Comparison of the best-obtained results from optimization algorithms in this thesis versus other studies on the same dataset.

Subject	**Our results**	**David Steryl et al.**	**Yang, Wang and Huang**
	EEGnet	**fbCSP+RF**	**EA-CSP-CNN**
S03	100.00%	96.67%	-
S09	93.80%	81.67%	-
S11	93.80%	63.33%	79.00%
S13	75.00%	88.33%	84.00%

The implementation of the three-class classifications showed that the flat model worked better than the hierarchical. In addition, it was shown that using EEGnet yielded better results than CSP+RF for both of the subjects S03 and S09. The highest accuracy score for subject S03 using EEGnet, optimal electrodes, and a flat model was 100.00%. For subject S09 the highest score was obtained using the same pipeline as subject S03, giving 87.50%.

The analysis of the results generally demonstrated that the classification and model-building process for MI tasks are highly subject-dependent. Consequently, it is challenging to draw any general conclusions that are applicable to all individuals performing MI. However, one of the few observations which were throughout for every subject was that the EEGnet outperformed all the ML-techniques. Thus, it seems that DL is a method worth focusing on.

6.3 Further Work

Many different methods for MI classification have been explored in this thesis. However, in order to ascertain a singular method that can effectively address all MI-related tasks and enable real-time testing of a drone, further research must be conducted.

Firstly, the NSGA algorithm should be used for optimization on multiple parameters at once, like electrodes and features. This is a very time-consuming job and has thus not been prioritized in this thesis. A way of making it less time-consuming is to add restrictions on the algorithm, which will be an evaluation needed to be done for further work. Another time-consuming experiment involves running all the NSGA algorithms on each individual subject, or multiple additional subjects. This way, a pattern across subjects might get easier to recognize, further leading to a subject-independent model.

Further, directing attention toward the development of subject-independent models and TL can lead to models which are less computationally expensive and capable of achieving high accuracies on subjects that have limited data available. This requires a closer examination of each individual subject's EEG pattern, with the objective of identifying which subjects to transfer knowledge from. Additionally, feature and electrode selection for subject-independent models might contribute to better models. To accomplish this, conducting both electrode selection and feature selection across all subjects combined must be done, to find the most optimal subject-independent subsets. Furthermore, for both ML and DL based TL models, employing an electrode selection on all subjects prior to the transfer could have potentially augmented the attained results.

The order of the experiments might have affected the results obtained. Consequently, reordering the sequence of experiments, such as performing electrode selection before pipeline selection, could potentially yield other results. Similarly, altering the sequence to conduct feature selection after electrode selection might also yield dissimilar results. Thus, as previously suggested, a combined approach encompassing both feature and electrode selections might be a more optimal selection strategy.

It is worth noting that fbCSP, a variant of CSP, has shown great results in MI classification as stated in the literature review. Therefore, the implementation and testing of fbCSP should be pursued to see if it can enhance the results compared to those obtained from CSP. Furthermore, filter-band optimization is also further work to be done, in addition to testing different parameters in the classifiers.

The electrode selection showed that subject S11 gets the highest accuracies when utilizing electrodes from different sections of the head than the other subjects. To reduce this difference among subjects, and to reduce the effect of the head shape and the shifted positions of electrodes, a step for further work could be testing a moving flexEEG for data collection[72].

Lastly, the live simulation was only tested on an already collected data stream. Thus, an important area for further work involves conducting experiments using an EEG recording helmet in conjunction with a live human subject. This approach would closely mimic real-world scenarios, particularly in contexts where the technology is intended for implementation in devices assisting individuals with paralysis or motor impairments. Furthermore, it is imperative to develop better models yielding higher accuracies. This advancement is important to make the classification safe and reliable for MI-based actuation of drones, wheelchairs, and similar applications.

References

[1] Dale Purves, George J. Augustine, David Fitzpatrick, William C. Hall, Anthony-Samuel LaMantia, James O. McNamara, and S. Mark Williams. *NEUROSCIENCE, THIRD EDITION*. Sinauer Associates, 2004.

[2] Bernhard Graimann, Brendan Z Allison, and Gert Pfurtscheller. *Brain-computer interfaces: Revolutionizing human-computer interaction*. Springer Science & Business Media, 2010.

[3] Swati Vaid, Preeti Singh, and Chamandeep Kaur. Eeg signal analysis for bci interface: A review. In *2015 fifth international conference on advanced computing & communication technologies*, pages 143–147. IEEE, 2015.

[4] Larry Squire, Darwin Berg, Floyd E Bloom, Sascha Du Lac, Anirvan Ghosh, and Nicholas C Spitzer. *Fundamenta Neuroscience, Third Edition*. Academic press, 2008.

[5] Li Hu and Zhiguo Zhang. *EEG signal processing and feature extraction*. Springer, 2019.

[6] Martin Lotze and Ulrike Halsband. Motor imagery. *Journal of Physiology-paris*, 99(4-6):386–395, 2006.

[7] Laura Leuchs. Choosing your reference–and why it matters. *Brain Products*, pages 03–05, 2019.

[8] Kip A Ludwig, Rachel M Miriani, Nicholas B Langhals, Michael D Joseph, David J Anderson, and Daryl R Kipke. Using a common average reference to improve cortical neuron recordings from microelectrode arrays. *Journal of neurophysiology*, 101(3):1679–1689, 2009.

[9] Dennis J McFarland, Lynn M McCane, Stephen V David, and Jonathan R Wolpaw. Spatial filter selection for eeg-based communication. *Electroencephalography and clinical Neurophysiology*, 103(3):386–394, 1997.

[10] Syahrull Hi Fi Syam, Heba Lakany, RB Ahmad, and Bernard A Conway. Comparing common average referencing to laplacian referencing in detecting imagination and intention of movement for brain computer interface. In *MATEC Web of Conferences*, volume 140, 2017.

[11] Luis Alfredo Moctezuma and Marta Molinas. Eeg channel-selection method for epileptic-seizure classification based on multi-objective optimization. *Frontiers in neuroscience*, 14:593, 2020.

[12] Amara Graps. An introduction to wavelets. *IEEE computational science and engineering*, 2(2):50–61, 1995.

[13] Kjersti Brynestad and Erlend Vatsåg. An asynchronous motor imagery-based brain-computer interface for two-dimensional drone control. Master's thesis, NTNU, 06 2021.

[14] Luis Alfredo Moctezuma and Marta Molinas. Classification of low-density eeg for epileptic seizures by energy and fractal features based on emd. *Journal of biomedical research*, 34(3):180, 2020.

[15] Dong Kyu Lee, Junyong In, and Sangseok Lee. Standard deviation and standard error of the mean. *Korean journal of anesthesiology*, 68(3):220–223, 2015.

[16] Asmaa Hamad, Essam H Houssein, Aboul Ella Hassanien, and Aly A Fahmy. Feature extraction of epilepsy eeg using discrete wavelet transform. In *2016 12th international computer engineering conference (ICENCO)*, pages 190–195. IEEE, 2016.

[17] Kazi Aminul Islam and Gleb V Tcheslavski. Independent component analysis for eog artifacts minimization of eeg signals using kurtosis as a threshold. In *2015 2nd International Conference on Electrical Information and Communication Technologies (EICT)*, pages 137–142. IEEE, 2015.

[18] Seung-Hyeon Oh, Yu-Ri Lee, and Hyoung-Nam Kim. A novel eeg feature extraction method using hjorth parameter. *International Journal of Electronics and Electrical Engineering*, 2(2):106–110, 2014.

[19] Rosana Esteller, G Vachtsevanos, J Echauz, and B Lilt. A comparison of fractal dimension algorithms using synthetic and experimental data. In *1999 IEEE International Symposium on Circuits and Systems (ISCAS)*, volume 3, pages 199–202. IEEE, 1999.

[20] Carlos Sevcik. On fractal dimension of waveforms. *Chaos Solitons and Fractals*, 28(2):579–580, 2006.

[21] R Esteller, J Echauz, and T Tcheng. Comparison of line length feature before and after brain electrical stimulation in epileptic patients. In *The 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, volume 2, pages 4710–4713. IEEE, 2004.

[22] Kai Keng Ang, Zheng Yang Chin, Haihong Zhang, and Cuntai Guan. Filter bank common spatial pattern (fbcsp) in brain-computer interface. In *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*, pages 2390–2397. IEEE, 2008.

[23] Nicklas Stubkjær Holm, Sadasivan Puthusserypady, et al. An improved five class mi based bci scheme for drone control using filter bank csp. In *2019 7th International Winter Conference on Brain-Computer Interface (BCI)*, pages 1–6. IEEE, 2019.

[24] Adele Cutler, D Richard Cutler, and John R Stevens. Random forests. *Ensemble machine learning: Methods and applications*, pages 157–175, 2012.

[25] O González-Recio, JA Jiménez-Montero, and R Alenda. The gradient boosting algorithm and random boosting for genome-assisted evaluation in large data sets. *Journal of dairy science*, 96(1):614–624, 2013.

[26] Petros Xanthopoulos, Panos M Pardalos, Theodore B Trafalis, Petros Xanthopoulos, Panos M Pardalos, and Theodore B Trafalis. Linear discriminant analysis. *Robust data mining*, pages 27–33, 2013.

[27] William S Noble. What is a support vector machine? *Nature biotechnology*, 24(12):1565–1567, 2006.

[28] Saad Albawi, Tareq Abed Mohammed, and Saad Al-Zawi. Understanding of a convolutional neural network. In *2017 international conference on engineering and technology (ICET)*, pages 1–6. Ieee, 2017.

[29] Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Methods for interpreting and understanding deep neural networks. *Digital signal processing*, 73:1–15, 2018.

[30] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[31] Navneet Tibrewal, Nikki Leeuwis, and Maryam Alimardani. Classification of motor imagery eeg using deep learning increases performance in inefficient bci users. *Plos one*, 17(7):e0268880, 2022.

[32] Navneet Tibrewal, Nikki Leeuwis, and Maryam Alimardani. Classification of motor imagery eeg using deep learning increases performance in inefficient bci users. *Plos one*, 17(7):e0268880, 2022.

[33] Rahim Barzegar, Mohammad Taghi Aalami, and Jan Adamowski. Coupling a hybrid cnn-lstm deep learning model with a boundary corrected maximal overlap discrete wavelet transform for multiscale lake water level forecasting. *Journal of Hydrology*, 598:126196, 2021.

[34] Vernon J Lawhern, Amelia J Solon, Nicholas R Waytowich, Stephen M Gordon, Chou P Hung, and Brent J Lance. Eegnet: a compact convolutional neural network for eeg-based brain–computer interfaces. *Journal of neural engineering*, 15(5):056013, 2018.

[35] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[36] Feng Liang, Zhichao Tian, Ming Dong, Shuting Cheng, Li Sun, Hai Li, Yiran Chen, and Guohe Zhang. Efficient neural network using pointwise convolution kernels with linear phase constraint. *Neurocomputing*, 423:572–579, 2021.

[37] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big data*, 3(1):1–40, 2016.

[38] Dipanjan Sarkar, Raghav Bali, and Tamoghna Ghosh. *Hands-On Transfer Learning with Python: Implement advanced deep learning and neural network models using TensorFlow and Keras*. Packt Publishing Ltd, 2018.

[39] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2010.

[40] Sourabh Katoch, Sumit Singh Chauhan, and Vijay Kumar. A review on genetic algorithm: past, present, and future. *Multimedia Tools and Applications*, 80:8091–8126, 2021.

[41] Chunteng Bao, Lihong Xu, Erik D Goodman, and Leilei Cao. A novel non-dominated sorting algorithm for evolutionary multi-objective optimization. *Journal of Computational Science*, 23:31–43, 2017.

[42] Rajnikant H Bhesdadiya, Indrajit N Trivedi, Pradeep Jangir, Narottam Jangir, and Arvind Kumar. An nsga-iii algorithm for solving multi-objective economic/environmental dispatch problem. *Cogent Engineering*, 3(1):1269383, 2016.

[43] Hamdi Altaheri, Ghulam Muhammad, Mansour Alsulaiman, Syed Umar Amin, Ghadir Ali Altuwaijri, Wadood Abdul, Mohamed A Bencherif, and Mohammed Faisal. Deep learning techniques for classification of electroencephalogram (eeg) motor imagery (mi) signals: A review. *Neural Computing and Applications*, pages 1–42, 2021.

[44] Yijun Wang, Shangkai Gao, and Xiaorong Gao. Common spatial pattern method for channel selelction in motor imagery based brain-computer interface. In *2005 IEEE engineering in medicine and biology 27th annual conference*, pages 5392–5395. IEEE, 2006.

[45] Mahnaz Arvaneh, Cuntai Guan, Kai Keng Ang, and Chai Quek. Robust eeg channel selection across sessions in brain-computer interface involving stroke patients. In *The 2012 International Joint Conference on Neural Networks (IJCNN)*, pages 1–6. IEEE, 2012.

[46] Muhammad Zeeshan Baig, Nauman Aslam, and Hubert PH Shum. Filtering techniques for channel selection in motor imagery eeg applications: a survey. *Artificial intelligence review*, 53:1207–1232, 2020.

[47] Swati Aggarwal and Nupur Chugh. Review of machine learning techniques for eeg based brain computer interface. *Archives of Computational Methods in Engineering*, pages 1–20, 2022.

[48] Natasha Padfield, Jaime Zabalza, Huimin Zhao, Valentin Masero, and Jinchang Ren. Eeg-based brain-computer interfaces using motor-imagery: Techniques and challenges. *Sensors*, 19(6):1423, 2019.

[49] Fabien Lotte, Laurent Bougrain, Andrzej Cichocki, Maureen Clerc, Marco Congedo, Alain Rakotomamonjy, and Florian Yger. A review of classification algorithms for eeg-based brain–computer interfaces: a 10 year update. *Journal of neural engineering*, 15(3):031005, 2018.

[50] Yong-Jeong Kim, No-Sang Kwak, and Seong-Whan Lee. Classification of motor imagery for ear-eeg based brain-computer interface. In *2018 6th International Conference on Brain-Computer Interface (BCI)*, pages 1–2. IEEE, 2018.

[51] Rongrong Fu, Yongsheng Tian, Tiantian Bao, Zong Meng, and Peiming Shi. Improvement motor imagery eeg classification based on regularized linear discriminant analysis. *Journal of medical systems*, 43:1–13, 2019.

[52] David Steyrl, Reinhold Scherer, Josef Faller, and Gernot R Müller-Putz. Random forests in non-invasive sensorimotor rhythm brain-computer interfaces: a practical and convenient non-linear classifier. *Biomedical Engineering/Biomedizinische Technik*, 61(1):77–86, 2016.

[53] Maouia Bentlemsan, ET-Tahir Zemouri, Djamel Bouchaffra, Bahia Yahya-Zoubir, and Karim Ferroudji. Random forest and filter bank common spatial patterns for eeg-based motor imagery classification. In *2014 5th International conference on intelligent systems, modelling and simulation*, pages 235–238. IEEE, 2014.

[54] David Steyrl, Reinhold Scherer, Oswin Förstner, and Gernot R Müller-Putz. Motor imagery brain-computer interfaces: random forests vs regularized lda-non-linear beats linear. In *Proceedings of the 6th international brain-computer interface conference*, pages 241–244, 2014.

[55] Fangzhou Xu, Xiaoyan Xu, Yanan Sun, Jincheng Li, Gege Dong, Yuandong Wang, Han Li, Lei Wang, Yingchun Zhang, Shaopeng Pang, et al. A framework for motor imagery with lstm neural network. *Computer methods and programs in biomedicine*, 218:106692, 2022.

[56] Sayeh Mirzaei and Parisa Ghasemi. Eeg motor imagery classification using dynamic connectivity patterns and convolutional autoencoder. *Biomedical Signal Processing and Control*, 68:102584, 2021.

[57] Zijian Wang, Lei Cao, Zuo Zhang, Xiaoliang Gong, Yaoru Sun, and Haoran Wang. Short time fourier transformation and deep neural networks for motor imagery brain computer interface recognition. *Concurrency and Computation: Practice and Experience*, 30(23):e4413, 2018.

[58] Zied Tayeb, Juri Fedjaev, Nejla Ghaboosi, Christoph Richter, Lukas Everding, Xingwei Qu, Yingyu Wu, Gordon Cheng, and Jörg Conradt. Validating deep neural networks for online decoding of motor imagery movements from eeg signals. *Sensors*, 19(1):210, 2019.

[59] Fatemeh Fahimi, Zhuo Zhang, Wooi Boon Goh, Tih-Shi Lee, Kai Keng Ang, and Cuntai Guan. Inter-subject transfer learning with an end-to-end deep convolutional neural network for eeg-based bci. *Journal of neural engineering*, 16(2):026007, 2019.

[60] Zhichuan Tang, Chao Li, and Shouqian Sun. Single-trial eeg classification of motor imagery using deep convolutional neural networks. *Optik*, 130:11–18, 2017.

[61] Zitong Wan, Rui Yang, Mengjie Huang, Nianyin Zeng, and Xiaohui Liu. A review on transfer learning in eeg signal analysis. *Neurocomputing*, 421:1–14, 2021.

[62] Jinpeng Li, Shuang Qiu, Yuan-Yuan Shen, Cheng-Lin Liu, and Huiguang He. Multisource transfer learning for cross-subject eeg emotion recognition. *IEEE transactions on cybernetics*, 50(7):3281–3293, 2019.

[63] Yuan-Pin Lin and Tzyy-Ping Jung. Improving eeg-based emotion classification using conditional transfer learning. *Frontiers in human neuroscience*, 11:334, 2017.

[64] Yuan-Pin Lin. Constructing a personalized cross-day eeg-based emotion-classification model using transfer learning. *IEEE journal of biomedical and health informatics*, 24(5):1255–1264, 2019.

[65] Ce Ju, Dashan Gao, Ravikiran Mane, Ben Tan, Yang Liu, and Cuntai Guan. Federated transfer learning for eeg signal classification. In *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pages 3040–3045. IEEE, 2020.

[66] David Steyrl, Reinhold Scherer, Oswin Förstner, and Gernot R Müller-Putz. Motor imagery brain-computer interfaces: random forests vs regularized lda-non-linear beats linear. In *Proceedings of the 6th international brain-computer interface conference*, pages 241–244, 2014.

[67] David Steyrl, Reinhold Scherer, Josef Faller, and Gernot R Müller-Putz. Random forests in non-invasive sensorimotor rhythm brain-computer interfaces: a practical and convenient non-linear classifier. *Biomedical Engineering/Biomedizinische Technik*, 61(1):77–86, 2016.

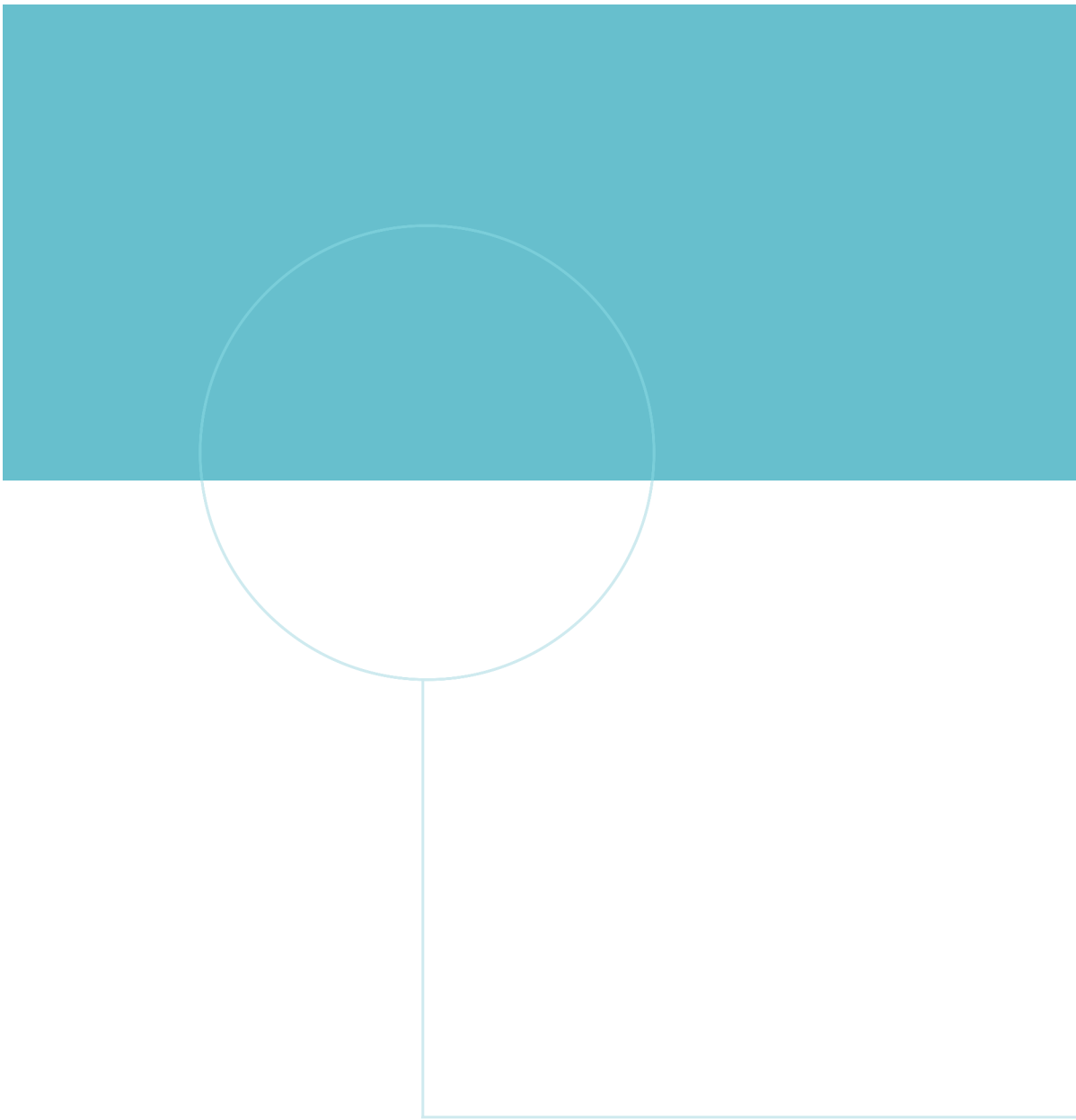
[68] Xuying Wang, Rui Yang, and Mengjie Huang. An unsupervised deep-transfer-learning-based motor imagery eeg classification scheme for brain–computer interface. *Sensors*, 22(6):2241, 2022.

[69] Hubert Cecotti and Girish Tiwale. Motor imagery classification combining riemannian geometry and artificial neural networks. In *Recent Trends in Image Processing and Pattern Recognition: 5th International Conference, RTIP2R 2022, Kingsville, TX, USA, December 1-2, 2022, Revised Selected Papers*, pages 179–189. Springer, 2023.

[70] Xia Chen, Xiangbin Teng, Han Chen, Yafeng Pan, and Philipp Geyer. Toward reliable signals decoding for electroencephalogram: A benchmark study to eegnex. *arXiv preprint arXiv:2207.12369*, 2022.

[71] Naomi Du Bois, Ryan Beveridge, Niall McShane, Tony Moore, and Damien Coyle. Signal quality assessment of a wearable electroencephalography (eeg) device built on a flexible printed circuit: Flexeeg. In *2022 IEEE International Conference on Metrology for Extended Reality, Artificial Intelligence and Neural Engineering (MetroXRAINE)*, pages 679–684. IEEE, 2022.

[72] Luis Alfredo Moctezuma, Andres Felipe Soler Guevara, Erwin Habibzadeh Tonekabony Shad, Alejandro Antonio Torres-Garcia, and Marta Molinas. David versus goliath: Low-density eeg unravels its power through adaptive signal analysis–flexeeg. In *4th HBP Student Conference On Interdisciplinary Brain Research*, 2020.



NTNU

Norwegian University of
Science and Technology