PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
Ministry of Higher Education and Scientific Research
University of Mohamed Khider – BISKRA
Faculty of Exact Sciences, Science of Nature and Life
Computer Science Department

N° d'ordre : IA21/M2/2022

Thesis

Submitted in fulfillment of the requirements for the Masters degree in

Computer science

Option : **Artificial Intelligence**

EEG Classification for Mind Controlling Applications using Multi-Method Approach

---

By:

TIBERMACINE IMAD EDDINE

Graduating on 28/06/2022 in front of the following committee of juries:

BOUREKKACHE Samir	MCA	President
NAPOLI Christian	Full Professor	Supervisor
TIBERMACINE Ahmed	MCA	Supervisor
BERGHIDA Meryem	MCA	Examiner

Session 2021-2022

الجمهورية الجزائرية الديمقراطية الشعبية

وزارة التعليم العالي والبحث العلمي

جامعة محمد خيضر بسكرة

تصريح شرفي

(خاص بالالتزام بقواعد النزاهة العلمية لإنجاز بحث)

أنا الممضي أسفله،

السيد (ة): .....تبرمسين عماد الدين.....الصفة: .....**طالب**.....

الحامل لبطاقة التعريف الوطنية رقم: 302080562 والصادرة بتاريخ: 15 جوان 2020

المسجل بكلية: العلوم الدقيقة و علوم الطبيعة و الحياة

قسم: الإعـــــــــــــــــــــــلام الآلـــــــي

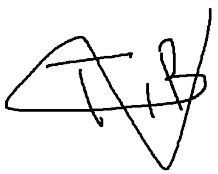
والمكلف بإنجاز مذكرة تخرج في الماستر عنوانها:

EEG Classification for Mind Controlling System Using Multi-Method Approach

أصرح بشرفي أني ألتزم بمراعاة المعايير العلمية والمنهجية ومعايير الأخلاقيات المهنية والنزاهة الأكاديمية المطلوبة في إنجاز البحث المذكور أعلاه.

التاريخ: 22 جوان 2022

توقيع المعني:



ABSTRACT

Since the first discovery of electroencephalography (EEG) principles in the 20's by Berger, scientists have used EEG signals in diagnosing brain conditions and other wide applications. In the last 2 decades, brain-computer interfaces (BCIs) and their technological advances have allowed people to use EEG for mind-controlling tasks, especially controlling robots by decoding and classifying EEG signals using Deep Learning (DL). However, anomalies in EEG signals induced by human abuses like alcohol and drugs or some diseases like Parkinson's, have made the classification task very hard. In this thesis, we have implemented a multi-method approach that uses two simultaneous models in order to generalize the motor imagery classification for mind-controlled robots from healthy patients to drug-addicted and alcoholic patients, and we discuss their applications to quad-rotors and wheeled mobile robots. We also accomplished the mind-controlling task in real time by accelerating our model's predictions. Finally, we realized the multi-robot controlling task, which would enable patients to control multiple robots (UAVs, wheeled mobile robots,... etc). The test results showed that the patients were able to use the proposed multi-method approach to control the mobile robot. The effectiveness of our study shows the high precision of attention-based Bi-LSTM compared to the SVM model, and by GCN compared to others, in classifying motor imagery EEG. The results were accurate and achieved the goals of the study. This will be a motivation to apply it to more complicated problems, like Parkinson's EEG studies.

Keywords: Brain-Computer Interface, Deep Learning, Machine Learning, Motor Imagery, Robot controllati dalla mente, Classification.

RÉSUMÉ

Depuis la première découverte des principes de l'électroencéphalographie (EEG) dans les années 20 par Berger, les scientifiques ont utilisé les signaux EEG dans le diagnostic des troubles cérébraux et d'autres applications larges. Au cours des deux dernières décennies, les interfaces cerveau-ordinateur (CEI) et leurs avancées technologiques ont permis aux gens d'utiliser l'EEG pour les tâches de contrôle mental, en particulier le contrôle des robots en décodant et en classant les signaux EEG à l'aide du Deep Learning (DL). Cependant, les anomalies dans les signaux EEG induits par les abus humains comme l'alcool et les drogues ou certaines maladies comme la maladie de Parkinson, ont rendu la tâche de classification très difficile. Dans ce mémoire, nous avons mis en œuvre une approche multi-méthode qui utilise deux modèles simultanés afin de généraliser la classification de l'imagerie motrice pour les robots contrôlés par l'esprit, des patients en bonne santé aux patients toxicomanes et alcooliques, et nous discutons de leurs applications àrotors et robots mobiles à roues. Nous avons également accompli la tâche de contrôle mental en temps réel en accélérant les prédictions de notre modèle. Enfin, nous avons réalisé la tâche de contrôle multi-robots, qui permettrait aux patients de contrôler plusieurs robots (UAV, robots mobiles à roues,... etc). Les résultats des tests ont montré que les patients ont pu utiliser l'approche multi-méthode proposée pour contrôler le robot mobile. L'efficacité de notre étude montre la haute précision du Bi-LSTM basé sur l'attention par rapport au modèle SVM, et par GCN par rapport les autres, dans la classification de l'EEG d'imagerie motrice. Les résultats étaient exacts et atteignaient les objectifs de l'étude. Ce sera une motivation pour l'appliquer à des problèmes plus compliqués, comme les études EEG de Parkinson.

Mots-clés: Interface cerveau-ordinateur, Apprentissage profond, Apprentissage automatique, Imagerie des moteurs, Robots contrôlés par le cerveau, Classification.

ASTRATTO

Dalla prima scoperta dei principi dell'elettroencefalografia (EEG) negli anni '20 da parte di Berger, gli scienziati hanno utilizzato i segnali EEG nella diagnosi delle condizioni cerebrali e in altre ampie applicazioni. Negli ultimi due decenni, le interfacce cervello-computer (CBI) e i loro progressi tecnologici hanno permesso alle persone di utilizzare l'EEG per compiti di controllo mentale, in particolare controllando i robot decodificando e classificando i segnali EEG utilizzando Deep Learning (DL). Tuttavia, anomalie nei segnali EEG indotte da abusi umani come alcol e droghe o alcune malattie come il Parkinson, hanno reso il compito di classificazione molto difficile. In questa tesi, abbiamo implementato un approccio multi-metodo che utilizza due modelli simultanei per generalizzare la classificazione delle immagini motorie per i robot controllati dalla mente da pazienti sani a pazienti tossicodipendenti e alcolisti, e discutiamo le loro applicazioni a quadrotori e robot mobili su ruote. Abbiamo anche completato il compito di controllo mentale in tempo reale accelerando le previsioni del nostro modello. Infine, abbiamo realizzato il compito di controllo multi-robot, che avrebbe permesso ai pazienti di controllare più robot (UAV, robot mobili a ruote,... ecc.). I risultati del test hanno dimostrato che i pazienti sono stati in grado di utilizzare l'approccio multi-metodo proposto per controllare il robot mobile. L'efficacia del nostro studio mostra l'elevata precisione dell'attenzione basata su Bi-LSTM rispetto al modello SVM, e da GCN rispetto agli altri, nella classificazione delle immagini motore EEG. I risultati sono stati accurati e hanno raggiunto gli obiettivi dello studio. Questa sarà una motivazione per applicarlo a problemi più complicati, come gli studi EEG di Parkinson.

Parole chiave: Interfaccia cervello-computer, Deep Learning, Machine Learning, Motor Imagery, Mind Controlled robots, classificazione.

ACKNOWLEDGEMENTS

God, the Almighty, is first and foremost praised and thanked for his showers of blessings that enabled me to successfully complete my studies. Professor Ahmed TIBERMACINE and Christian NAPOLI, my research supervisors, for giving me the opportunity and providing invaluable guidance throughout this work; to Professor Gaspare GALATI and Doctor Samuele RUSSO for their efforts and collaboration to publish some scientific papers; and to robotic full Professors Alessandro DE LUCA, Giorgio GRISETTI, Giuseppe ORIOLO, and Marilena VENDITELLI for the solid background in robotics. Their dynamism, vision, passion, and drive have profoundly motivated me. It was a real honor and privilege to work under their direction. I am appreciative for his assistance and patience during the entire process. I would want to thank all of the teachers in the Computer Science Department at Biskra who assisted me with my education. Thank you to the ERASMUS+ Office, Mr. Mohamed MOUMMI, Professor Mahmoud DEBABECHE, and Ester LATINI for the incredible opportunity to participate in an exchange program at Sapienza University, and to Professor Okba KAZAR for the chance to join the LINFI Laboratory and all of its members. Professor Ahmed BOUTERFAIA, the head of Biskra University, and Mr. Smail GUETTALA, for their support over the past two years, have my deepest gratitude. In addition, I would like to thank the Minister of Fisheries, Professor Hichem SALAOUATCHI, for his tremendous efforts to help me go further. I am indebted to my parents for their ongoing and unwavering love, prayers, care, and sacrifices for my education and future preparation. Additionally, for standing by me despite all the hurdles I experienced. They have provided me with the opportunities and experiences that have shaped my identity. My voyage would not have been possible without them, so I dedicate this achievement to them and to the soul of my beloved grandmother. My deepest appreciation goes to my brothers Assem, Nacer, Mourad, Youcef, and Hamda, my sisters Randa and Sophia, my uncle Professor Abdrrehman TIEBRMACINE, and

all my family members for always being by my side. I am appreciative of the assistance they've provided. I'd also like to acknowledge the affection of my nephews and nieces. Last but not least, I would like to express my gratitude to my friends and loved ones, especially Aymen, Houssem, Fouad, Nacer, ElBacher, Badreddine, Rahim, Moncef, Didine, Ahmed, Taha, TakiEddine, Hakim, Saad, Mohamed, Habib, Ilyes, Amine, Olga, Rania, Amani, Roufeida, Farah, Safinez, and everyone else, for their companionship and support.

Papers Submitted and Under Preparation

• Imad Eddine TIBERMACINE, Christian NAPOLI, Ahmed TIBERMACINE,Gaspare GALATI, Samuele RUSSO, "Mind controlled robots: Study case of drug addicted people using multi-method approach", *Submitted to MDPI Sensors*, ACCEPTED.

• Imad Eddine TIBERMACINE, Ahmed TIBERMACINE, Christian NAPOLI,Gaspare GALATI, Youcef YAHI, "GCNN-based Multi-method for the Classification of Motor Imagery for Drug Addicted subjects.", *Submitted to MDPI Sensors*, June. 2022.

• Imad Eddine TIBERMACINE,Christian NAPOLI, Ahmed TIBERMACINE, Gaspare GALATI, "A Survey of Motor Imagery Classification using Revolutionary Deep Learning Methods", *Under Preparation/Elsevier Biomedical Signal Processing and Control*.

• Imad Eddine TIBERMACINE, Ahmed TIBERMACINE, Christian NAPOLI, "Real-Time multi robot controlling via EOG and Motor Imagery Classification", *Under Preparation/MDPI Sensors*.

• Imad Eddine TIBERMACINE, Christian NAPOLI, Ahmed TIBERMACINE, Gaspare GALATI, Samuele RUSSO,"EEG Motor Imagery DataSet for healthy and drug-addicted patients", *Under Preparation/IEEE*.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Introduction

Since Berger's first discovery of electroencephalography (EEG) principles in the 1920s, scientists have employed EEG signals in diagnosing brain diseases and in a variety of other contexts. In the past two decades, brain-computer interfaces (BCIs) and associated technological advancements have enabled humans to use EEG for mind-controlled tasks, particularly for directing robots by decoding and classifying EEG signals using Deep Learning (DL). However, abnormalities in EEG signals caused by human abuses such as alcohol and narcotics, as well as certain disorders such as Parkinson's, have made classification extremely difficult. In this dissertation, we have implemented a multi-method approach that employs two simultaneous models to generalize the motor imagery classification for mind-controlled robots from healthy patients to drug-addicted and alcoholic patients, and we discuss their applications to quad-rotors and wheeled mobile robots. We also completed the mind-controlling task in real time by speeding the predictions of our model. Finally, we implemented the multi-robot control task, allowing patients to operate multiple robots (UAVs, wheeled mobile robots,... etc). The test findings demonstrated that the patients were able to control the mobile robot using the proposed multi-method strategy. In identifying motor imagery EEG, our study demonstrates the superior accuracy of attention-based Bi-LSTM over the SVM model, and of GCN over other algorithms. The results were accurate and accomplished the study's objectives. This will encourage its use to more complex situations, such as EEG investigations for Parkinson's disease.

In the first chapter, we provide an overview of relevant research on Brain-Computer Interfaces and its applications in robot control. The first section of this chapter describes how BCIs function, covering the key brain regions and data collection methods. The most prominent electroencephalographic (EEG) markers are then described. The second section examines prior and ongoing BCI-based robotics development and projects. The concluding portion focuses on the contributions of our research to the advancement of BCIs for mind-controlling robots.

In the second chapter, we discuss the background material relevant to the design and implementation of a brain-computer interface, including a review of the various recording and signal processing techniques required to produce a usable signal as well as a discussion of BCI classification techniques. In order for you to comprehend our work, we will describe each phase in greater detail in light of the BCI's big picture.

In the third chapter, we describe the use of machine learning in the preprocessing and categorization of EEG signals. In addition, we cover numerous machine learning and deep learning classification algorithms for EEG inputs. In addition, we describe how ML/DL and its hyperparameters will be utilized for EEG classification and decoding.

In the final chapter, we describe the algorithms, datasets, hardware, and software that were used to generate the subsequent conclusions. In addition, we describe the numerous technical components of our system's deployment and execution. We explain the acquisition of data from numerous local individuals, the conditions under which it was collected, and the EEG headgear that was utilized. Then, we discuss our proposed model and its flowchart, describing its components and the techniques employed (preprocessing, artifact removal, feature extraction, classification, etc.). Lastly, we will emphasize several robotic notations to ease our procedure.

In conclusion, we describe a novel multi-method approach that employs simultaneous models including SVM, attention-based Bi-LSTM, and GCN to classify motor imagery tasks for healthy and sick (drugged, drunken) patients. The results of our tests demonstrated that the suggested method significantly outperformed other methods that have been applied to solely healthy patients in classifying human judgments to operate robots. It demonstrates that our model with an attention mechanism performed an excellent job of identifying and differentiating EEG patterns that were distinct due to the effects of drugs on the brain, but the most accurate models were the gcn, which obtained very respectable results. However, we were able to accelerate models and control robots in real time. Lastly, we discussed the obtained results in two studies: an evaluation study and a comparative study.

Chapter 1

Theoretical Background

This chapter provides an overview of related research in the subject of Brain-Computer Interfaces and its implementations in the field of robot controlling. The first section of this chapter explains how BCIs work, including what the primary brain structures are and how brain data is collected. Following that, the most prevalent ElectroEncephaloGraphy (EEG) markers are discussed. The second section surveys the previous and current works and projects in robotic fields that are based on BCIs.The final section deals with the contributions of our work in order to enhance the usage of BCIs for mind-controlling robots.

1.1 Overview

Controlling a virtual world with one's brain activity has long been considered by researchers and artists. Examples of such interfaces can be found in science fiction. For example, in the film "X-Men," "Professor-X" has the ability to control everything with his mind alone. In "The Matrix," humans have a connection port in the back of their heads that allows them to connect to and interact with the Matrix, a virtual world. In another 2011 film, "Source Code," a soldier is embodied into a deceased person during her final 8 minutes in order to identify a bomber. This soldier is linked to a Brain-Computer Interface (BCI), which allows him to view and interact with an alternate reality. These examples pave the way for new applications of BCIs, particularly for controlling virtual or real worlds[17]. The brain-computer interface (BCI) is a communication and control channel that "acquires brain signals, analyzes them, and converts them into commands that are conveyed to an output device to carry out a desired action" and "does not rely on the brain's usual output pathways in any way"[18]. BCIs

Figure 1.1: Professor-X in X-Men

are an intriguing method of controlling devices[19]. After several decades of research and several advancements, notably in recent years, we are still a long way from using BCI-applications for everyday chores[20]. The majority of their uses are in medicine and research[21]. One reason for this is that their capabilities are still highly limited compared to popular human device interfaces such as button-based, touch-based, voice-recognition-based, and gesture-recognition-based interfaces[22]. BCIs, on the other hand, have one very appealing feature: they are the only known human device interfaces that do not require muscle movements. As a result, BCIs are primarily beneficial to people who have lost control of all muscles that could otherwise be used for communication[23]. BCI systems come in a variety of flavors, and multiple technologies can be utilized to gather neurophysiological data from the brain. Functional Magnetic Resonance Imaging (fMRI), functional Near Infrared Spectroscopy (fNIRS), ElectroEncephaloGraphy (EEG), MagnetoEncephaloGraphy (MEG), and ElectroCorticoGraphy are examples of these technologies (ECoG)[24][25][26]. Brain signal analysis necessitates some understanding of the human brain's inner workings. This part gives an overview of brain structure concepts, as well as how to execute brain data acquisition and interpret the results. EEG is the most often used method among all of them since it is non-invasive and gives excellent temporal resolution in terms of recorded brain activity[27].

1.2 Definitions of BCI

1.2.1 Synchronous or Asynchronous BCI

Either synchronous or asynchronous BCIs are possible. When operating a synchronous BCI, the user has no control over when a command is received, and must instead arrange their actions to correspond with the BCI's timing in order to achieve optimal performance; for instance, minimizing blinking and muscular movements prior to the active time. Asynchronous BCIs, also known as self-paced BCIs, allow the user to control the speed of the BCI and stop sending orders if desired. Asynchronous BCIs are technically more difficult to implement than synchronous BCIs. This is mostly owing to difficulty in recognizing the brain's so-called "idle" state, a misnomer as the brain is always functioning. As a result, it is challenging to determine whether a user does not wish to send a command. However, several methods have been developed to achieve this asyn- chronicity, including: thresholding, where commands are only sent when the user's brain activity exceeds a predetermined threshold; using a so-called 'brain-switch,' where a neural command is used to start or stop the BCI from sending commands; and using muscle movements or blinks as a trigger to initiate BCI functions[28].

1.2.2 Invasive and Non-invasive BCI

Based on whether or not the BCI penetrates the skin, it can be classified as invasive or non-invasive. Invasive BCIs assess local field potentials (LFP) directly by piercing brain tissue[29], whereas noninvasive BCIs monitor scalp potentials from the head's surface. More information regarding invasive and non-invasive methods can be found in the following chapter.

1.3 Neurophysiological Foundations of BCI

1.3.1 Brain Structure

The brain stem, cerebellum, and cerebrum are the three sections that make up the human brain. Left and right hemispheres divide the cerebrum. The cortex, which covers the cerebrum's surface, is divided into four "lobes."

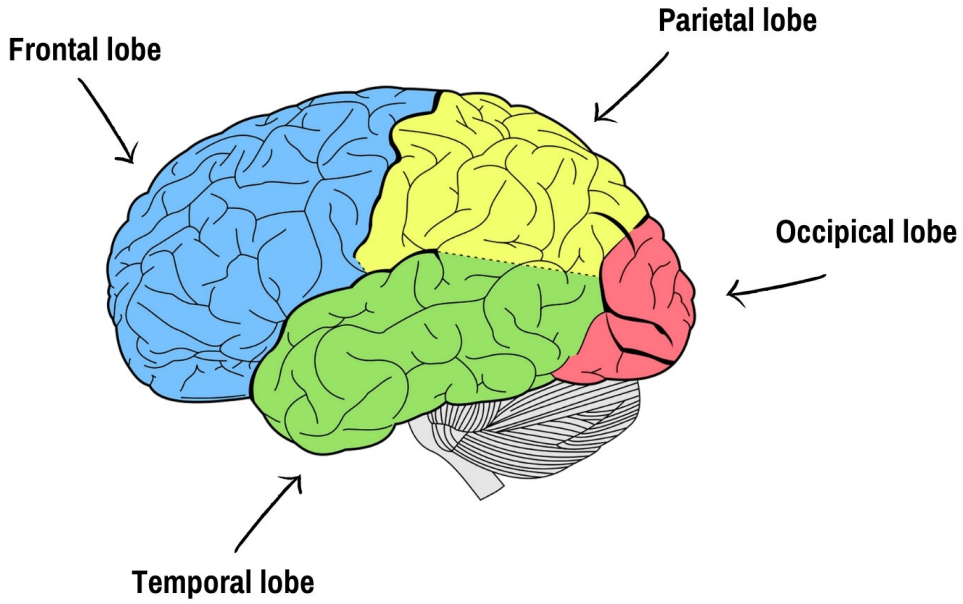


Figure 1.2: Brain Lobes.

The central sulcus separates the frontal lobe from the parietal lobe, and the lateral sulcus separates the frontal lobe from the temporal lobe. Higher executive skills such as emotional regulation, planning, reasoning, and problem solving, as well as cognitive processes such as speech and movement, are all performed in the frontal lobe. The central sulcus separates the parietal lobe from the frontal lobe. Sensory integration, including touch, warmth, pressure, and pain, as well as reading/writing, language comprehension, concentration, and spatial awareness, are all handled by areas in the parietal lobe. The temporal lobe, which is separated from the frontal lobe by the lateral fissure, comprises sensory-processing regions that are critical for hearing, language recognition, and memory formation. Finally, the occipital lobe is the brain's primary

visual processing center. Visual information from the eyes is received by the primary visual cortex, often known as V1. This data is sent to numerous secondary visual processing areas, which evaluate depth, distance, location, and object identity[30].

Electrical fields, blood pressure, and magnetic fields can all be used to record brain activity. The process of gathering this data is known as ”acquisition.” Depending on whether the measurement equipment is put on the brain tissues directly or on the scalp, retrieval of brain data can be accomplished via invasive or non-invasive procedures[31]. Signal acquisition techniques, which can be invasive or non-invasive, and signal evocation methods, which can be exogenous or endogenous, distinguish BCIs (the way a subject is stimulated to create the desired signals). Invasive systems are less susceptible to noise than non-invasive interfaces, but they need a surgical procedure to implant and must be removed or replaced after a certain period of time due to the rejection phenomena [32]. Non-invasive BCIs can use EEG[27], near-infrared spectroscopy[33], magnetoencephalography[34], and functional magnetic resonance imaging[35], while invasive BCIs mostly use electrocorticography[36]. In the non-invasive technique the sensors are placed on the scalp to measure the electrical potentials produced by the brain (EEG) or the magnetic field (MEG) while the micro-electrodes are inserted into the brain and used to measure the activity of a single neuron. The exists another technique named semi-invasive, where the electrodes are placed on the exposed surface of the brain(ECoG).

EEG is used in many BCI systems because it is simple to utilize and inexpensive compared to magnetoencephalography or functional magnetic resonance imaging[27]. Because of the electric potentials it emits, EEG uses active or passive electrodes placed on the scalp to receive data from the brain. The EEG electrodes are usually linked to a cap and put on the surface of the head[27]. To improve the conductivity between the skull and the electrodes, gel or salty water is employed. This means that any EEG system will take a certain amount of time to set up (typically 15 minutes). The 10-20 system is often used to put electrodes. There exist another EEG electrodes placement system named 10-10 system and it contains basically 13 electrodes, but this last one is

rarely used and that's because of the high acquisition the first system gives.

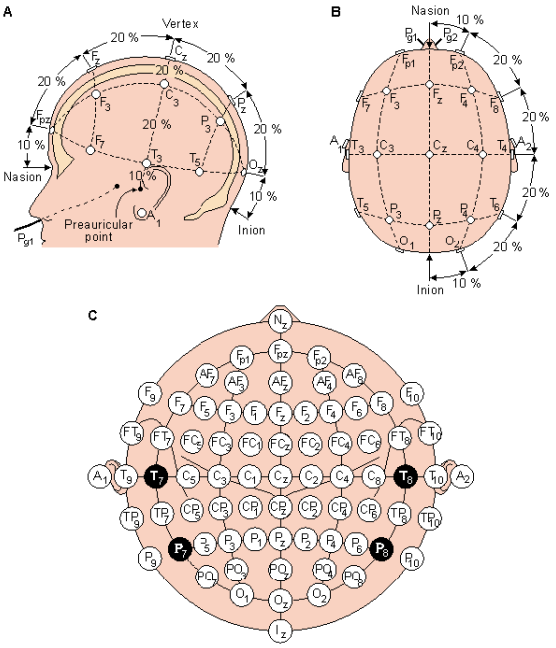


Figure 1.3: EEG Electrodes Placement 10-20 International System

1.3.2 Electroencephalography Rhythms

Rhythms are oscillations which can be seen in specific areas of the brain and at specific frequency bands. The recorded wave forms reflect the cortical electrical activity. They can provide some information about a person's mental state and can be voluntarily controlled[37]. These rhythms are alpha, beta, gamma, delta, theta and mu. But generally, only 4 periodic rhythms are used; alpha, beta, delta, and theta. The previous rhythms are identified by frequency (Hz or cycles/sec) and amplitude (generally micro-volts uV).

- Alpha waves (8-13Hz) are found in the posterior part of the brain and are stronger on the non-dominant side. It is most noticeable when the eyes are closed or in a relaxed state: increased alertness reduces its amplitude.

- When the user is awake and conscious, the beta rhythm (13-30Hz) can be detected and it is affected by movement.

- The gamma rhythm (greater than 30Hz) is associated with cognitive and motor activity but is difficult to detect using scalp-based EEG.
- The delta rhythm (1-4Hz) is mostly detected while sleeping or deep sleeping (adults).
- The theta rhythm (4-8Hz) can be seen in young children during sleep and is known to form a spike when the user is attempting to suppress a response or action[38].
- The mu rhythm (8-13Hz) is motor-based and varies depending on how the user moves[39].

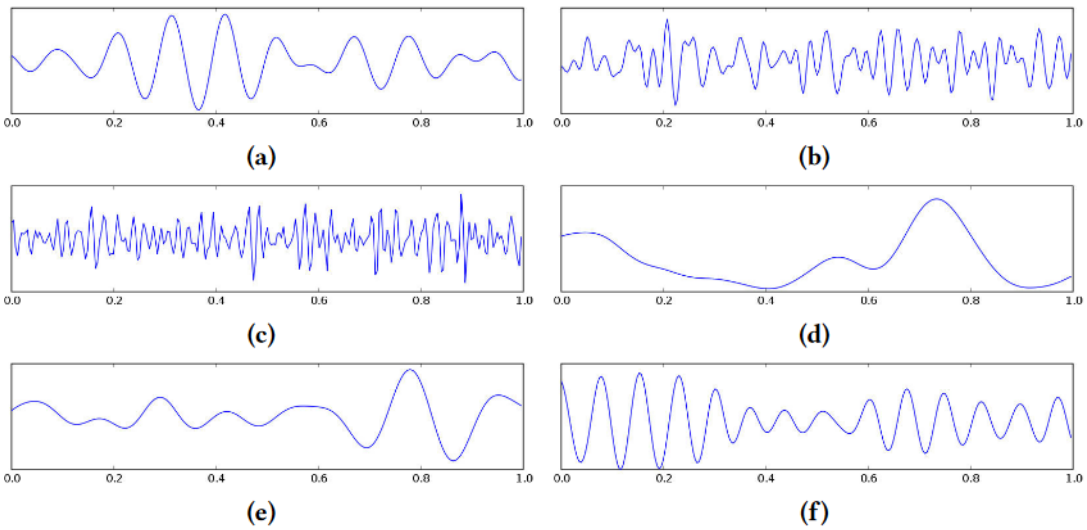


Figure 1.4: EEG Rhythms

1.3.3 Effects of Drugs on EEG

Despite the fact that excessive beta activity and a slight theta rise may be the most typical EEG changes linked with medication, more spectacular changes may sometimes occur. Although abnormalities such as diffuse delta, triphasic waves, bisynchronous spikes or polyspikes, burst suppression, or electrocerebral inactivity are often indicative of a poor prognosis, if drug administration is the only or primary reason, these patterns may revert to a normal EEG[40].

1.3.4 Brain Imagine Techniques

In this section, we provide a quick overview of typical non-invasive brain imaging techniques and more fundamental information concerning non-invasive brain signal detection (e.g. concepts, characteristics, advantages, and draw- backs). Figure 1.5 depicts a classification of non-invasive brain signals based on the manner of signal collection. Noninvasive signals are classified as EEG, fNIRS, functional magnetic resonance imaging (fMRI), and magneto-encephalography (MEG)[41]

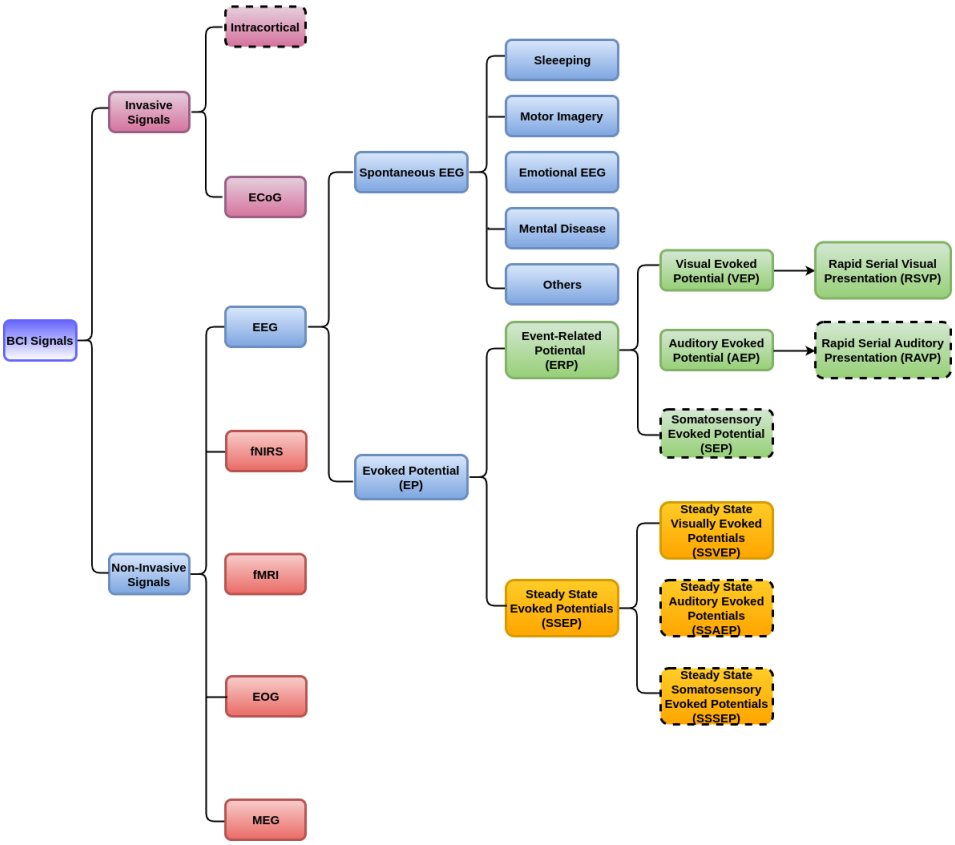


Figure 1.5: Classification of noninvasive brain signals. The dashed quadrilaterals (RAVP, SEP, SSAEP, and SSSEP) are omitted from this review because there is no prior research that focuses on deep learning techniques for them. P300, a positive potential recorded roughly 300 ms after the commencement of given stimuli, is not included in this signal tree because it is included by ERP (which refers to all the potentials after the presented stimuli). Other brain imaging techniques outside EEG (e.g., MEG and fNIRS) might theoretically include visual/auditory tasks in this categorization (e.g., MEG and fNIRS), however we removed them because there is no prior work employing deep learning on these tasks[1].

The second capability describes the properties of distinct brain signals. This survey focuses primarily on EEG signals and their subtypes, as they predominate among non-

Table 1.1: Summary of non-invasive brain signals' characteristics[1]

Signals	EEG	fNIRS	fMRI	MEG
Spatial resolution	Low	Intermediate	High	Intermediate
Temporal resolution	High	Low	Low	High
Signal-to-Noise Ratio	Low	Low	Intermediate	Low
Portability	High	High	Low	Low
Cost	Low	Low	High	High
Characteristic	Electrical	Metabolic	Metabolic	Magnetic

invasive signals. EEG measures the voltage fluctuations created by an electrical current within the neurons of a human subject. The electrodes linked to the scalp can measure numerous EEG signal types, including spontaneous EEG [42] and evoked potentials (EP) [43]. Depending on the context, spontaneous EEG subdivides further into sleep EEG, motor imagery EEG, emotional EEG, and mental disorder EEG, among others. According to the frequency of external stimuli, EP is also subdivided into ERPs [44] and steady-state evoked potentials (SSEPs) [45]. Based on the sorts of external inputs, each potential has visual, auditory, and somatosensory components.

1.3.5 Event-Related Desynchronisations/Synchronisations

Changes in SMRs that accompany imagined or actual motor tasks are known as event-related desynchronisations (ERD) and event-related synchronisations (ERS). The ERD is a drop in power in the upper alpha (mu) band and lower beta band, which occurs in the contralateral hemisphere around two seconds before movement beginning and becomes bilaterally symmetrical just before movement onset[46]. The ERS is an increase in strength that occurs following the completion of a motor task. ERS can also occur concurrently with ERD, albeit in a distinct cortical region[47]. ERD/ERS are generated topographically relative to the homuncular organization of the sensory cortices (Fig. 1.6). This indicates that ERD and ERS activity associated with foot movement will be most evident in the foot portion of the sensorimotor cortex. The homuncular organization also makes it difficult for existing non-invasive BCIs to differentiate between movements of the two feet, which are placed close together (roughly between the brain's medial longitudinal fissure), and between individual finger motions. Due

to the great distance between foot, right-, and left-hand movements, they are able to differentiate among them.

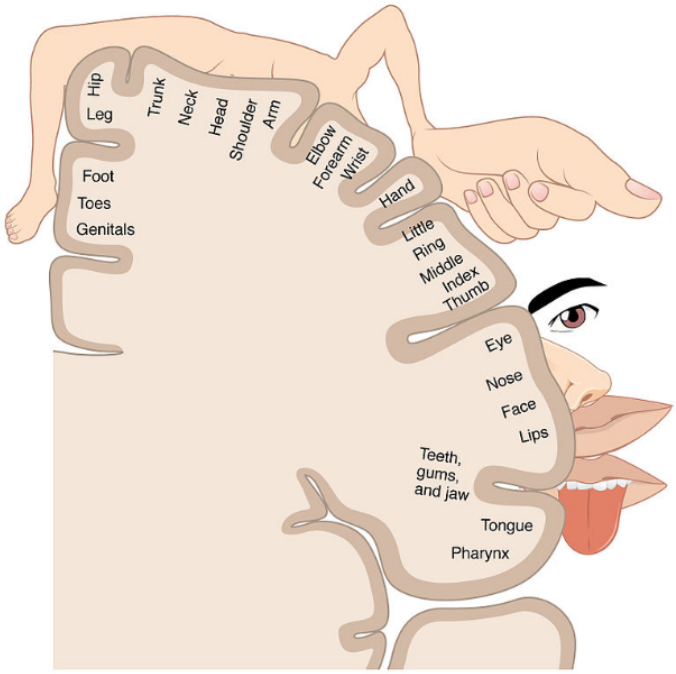


Figure 1.6: A 2-D cortical sensory homunculus, by OpenStax College (licensed under CC BY 3.0)[2]

1.3.6 Event-related potentials

Some events, such as sensory stimuli, motor activity, and memory challenges, have been observed to have a direct effect on the rhythms outlined in the previous section[46]. A rise in rhythmic activity is known as a ”event-related synchronization”, whereas a decline is known as a ”event-related desynchronization”. Event-related desynchronizations can occur milliseconds or even seconds before a movement is done or envisioned. For instance, picturing a right hand movement will result in a desynchronization of the mu and beta rhythms in the left motor cortex, which is event-related[5].

A paradigm known as ”motor imagery” employs event-related desynchronization, which relies mostly on the alpha and beta rhythms. Motor imagery depicts imagined motor activity, while motor execution depicts actual motor activity[5]. After the execution or imagining of a motor activity, it is possible to detect an alpha- or beta-rebound.

Motor imagery enables users to send commands to a brain-computer interface (BCI) by picturing making a movement[42].

The P300 is a positive waveform that appears 300 milliseconds after a rare and significant stimulation[48]. Commonly, the P300 paradigm is used to a grid of letters known as a ”P300 speller”[49]. This grid contains letters that may be black or light (ashing). The user must concentrate on one of the letters and mentally count the number of times it has been ashed. This allows the algorithm to determine which letter is focused. This paradigm has been applied successfully to paraplegic individuals, allowing them to converse ”using thoughts”[50].

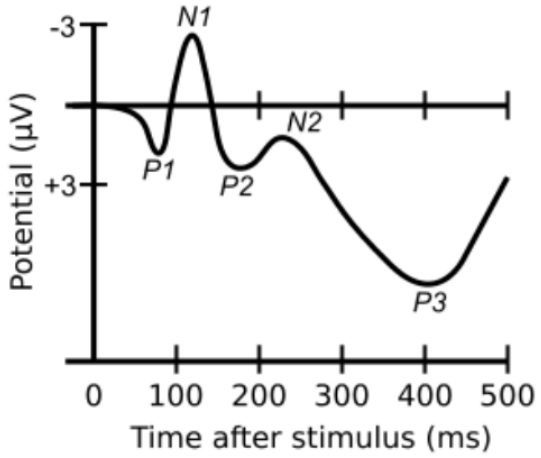


Figure 1.7: Illustration of Event-Related Potentials[2]

P300 evoked potential

A P300 evoked potential (also known as P3b) is an example of an ERP that consists of a positive peak reported in the parietal cortex 300ms after stimulus initiation. It was found by Sutton et al.[51] and is triggered when improbable events occur between highly probable events. The primary use of the P300 in BCI research is the ’P300 speller’[52], a grid of letters (usually 66) from which a user can select specific characters by concentrating their gaze on them. The letters flash randomly, but the BCI is time-locked to these flashes (referred to as the ”oddball paradigm”[53]) and letter selection is based on P300 wave production. Guger et al.[54] evaluated the usage of

the P300 signal for spelling and found an overall classification accuracy of 91%. In addition, 72.8% of subjects were able to spell sentences with perfect accuracy using the P300 signal, while less than 3% displayed full BCI impairment with the P300. The primary benefits of adopting a P300-based BCI are the low training times and great accuracy. The primary downside is its slowness.

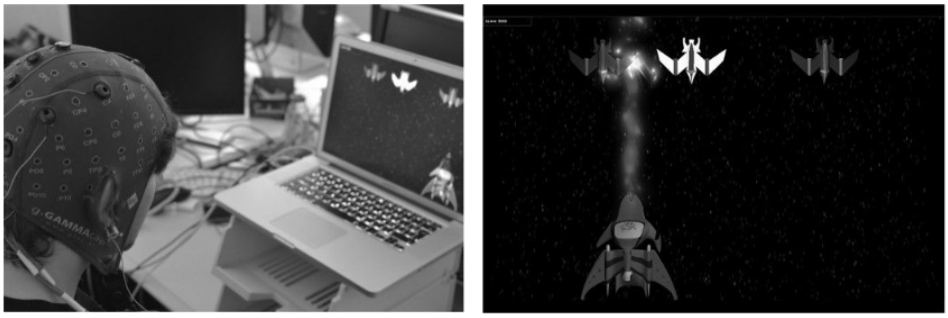
Error-related Potentials

Error-related potentials (ErrP) [55] refer to the brain's response to a user identifying an error, which can be utilized to control a BCI, typically for automatic error correction.

1.3.7 Steady State Visually Evoked Potential

The steady-state visually evoked potential (SSVEP) is a kind of the visually evoked potential (VEP), which is a brain response produced by a visual input. The SSVEP is phase-locked and induced by a repeated visual stimulus (RVS) such as a flickering light[56], or a reversing pattern[57], and becomes'steady' when the stimulus presentation rate exceeds a particular frequency. SSVEP responses are primarily detected by electrodes implanted above the occipital and parietal lobes[58], and have a spatial structure resembling a wave[59] with frequency characteristics identical to the triggering input. The RVS commences the selection of a cortical network that can oscillate at the same frequency[60], which means that the response has a high signal-to-noise ratio compared to the input. Popular because of its quick training period, high classification rate[60], and ability to be detected using non-invasive neuroimaging techniques such as EEG. They have been utilized in a variety of BCI types, including exoskeletons operated by BCI[61], [62], wheelchairs [62], [63], and robotic humanoids [64]. Using various techniques, the objective is to maximize the signal-to-noise ratio (SNR), as is the case with all other signal generating methods. By analyzing the brain impulses for specific frequencies related to the RVS frequency, researchers may determine which stimulus is being observed. While the brain is capable of producing SSVEP responses at RVS frequencies ranging from 1 to 90 Hz[56], optimum stimulation frequencies are

reported between 5.6 and 15.3 Hz, with the highest response occurring at 12 Hz[57]. Bakardjian et al.[57] stated that choosing between eight commands resulted in a mean classification accuracy of 98% (96-100%) and a mean command recognition delay of 3.4s (2.5-4.2s). The primary benefit of an SSVEP-based BCI is its high signal-to-noise ratio (SNR) and hence its excellent accuracy. The primary downside is that seeing the RVS poses a risk of seizures[65], and protracted viewing sessions might produce user weariness[66].

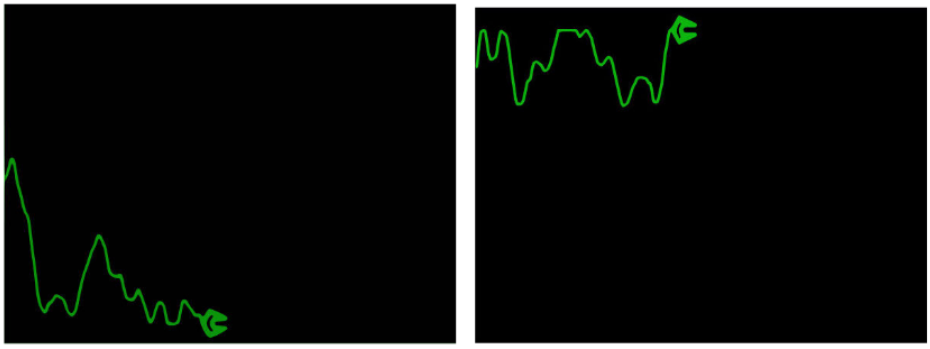


(a) (b)

Figure 1.8: A system that uses SSVEP to pilot a spacecraft. a) A user observing a computer screen with a game-like interface. (b) A spaceship operated by SSVEP and three targets that must be destroyed[3].

1.3.8 Concentration and relaxation mental states

Additionally, a BCI can be utilized to measure mental status. For instance, George et al.[4] devised a system for measuring the level of concentration and relaxation. Using ratios or combinations of the alpha, beta, and theta rhythms, this concentration level can be determined[**?**][67][68]. These mental states can also be classified using machine learning techniques[69].[4]George et al. contrasted these two strategies and concluded that the machine learning methodology performed better. In addition, they have discovered the optimal number and positioning of EEG electrodes on the skull. Their user study was based on a basic game-like 2D environment in which users moved an ascending or descending plane based on their mental state.



(a) Relaxation mental state. (b) Concentration mental state.

Figure 1.9: A straightforward 2D application that displays the concentration/relaxation mental state detected by a BCI[4].

1.4 Hybrid Brain-Computer Interfaces

A hybrid BCI (hBCI) is a system that combines at least two types of input signals, at least one of which is a BCI signal. The other signal or signals can come from: another BCI signal from the same modality, creating a system known as a 'pure' hBCI; another BCI signal from a different modality, such as combining EEG and fMRI; physiological signals such as heart rate or EMG, although whether using EMG constitutes a true hBCI is debatable; or from an intelligent device such as an eye tracker or intelligent wheelchair. Sequential hBCIs execute a function from a single modality at a time, whereas simultaneous hBCIs conduct actions from numerous modalities simultaneously. Numerous combinations of hBCIs, including P300 and SSVEP[70], ERD and SSVEP[71], and ERD and P300[72], have been shown to be effective. The primary benefit of hBCIs is their capacity to compensate for shortcomings in the modality or modalities. hBCIs can improve upon normal BCIs by increasing the available degrees of control (DoC)[71], raising the ITR[73], and enhancing accuracy[70].

A hybrid BCI system's components can function sequentially or simultaneously. Figure 1.8 depicts two hybrid BCIs (B and C) working concurrently and five operating sequentially (A, D- G). The sequential processing displays the "switch" and "selection" modes, which may be used, respectively, to activate the second component or choose between two additional components. The "+" sign represents the "fusion" procedure, indicating that the information received by various components is "added" to

produce the final output. This figure also depicts the many components of a hybrid BCI, including a non-EEG BCI device employed in E, an electrocardiography device in B and D, and an eye-tracking device in G. These devices will be discussed in the sections that follow[5].

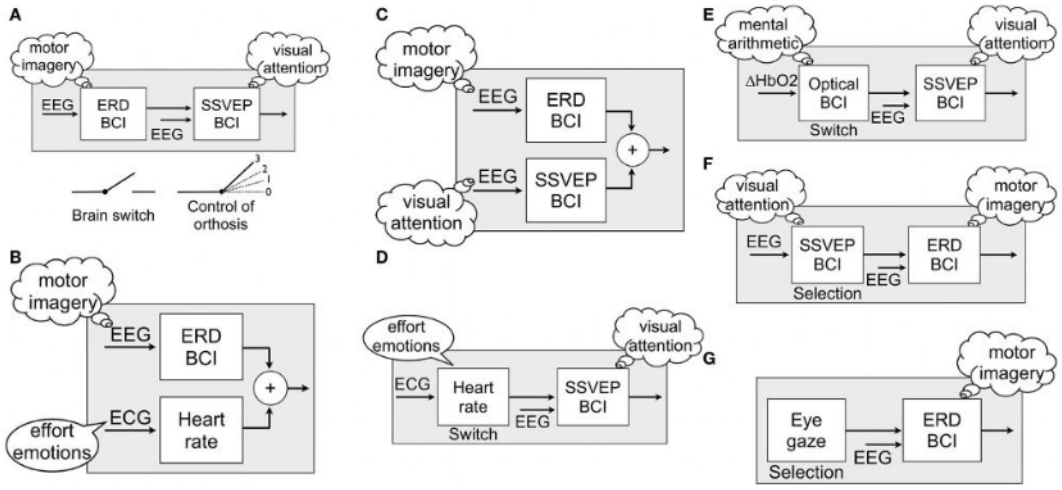


Figure 1.10: Illustrations of hybrid BCI components running sequentially or concurrently[5].

1.5 Principal ideas behind hybrid brain-computer interfaces

A hybrid brain-computer interface consists of numerous components that are interconnected to build a system. When BCIs are often associated, they are ”active” as opposed to ”passive,” meaning that users actively submit commands through the interface as opposed to being ”monitored” by it. The majority of BCI setups are active in the sense that the user must consciously manage her brain activity in order to trigger an event or accomplish a goal[74]. The data transfer rate of active BCIs is often below 25 bits per minute[75]. Passive BCIs are commonly employed in conjunction with other inputs and, as a result, are regularly included into hybrid BCIs. This section begins with an overview of the nature of the hybrid BCI’s constituent elements: ”mixed” (a BCI integrated with other inputs) or ”pure” (only BCIs). The second element of this section will discuss the sequentiality of data processing, such as whether data from one component is handled before or simultaneously with data from another component[5].

1.5.1 Brain-computer interfaces that are a mix or a pure hybrid

There are two categories of hybrid brain-computer interface systems, depending on whether they contain exclusively BCI devices or a combination of BCI and non-BCI equipment. Pure hybrid BCI refers to BCI systems that utilize solely BCI devices with a single or several paradigms. Mixed hybrid BCI systems often employ electrooculography, electromyography, and electrocardiography in addition to additional components[5].

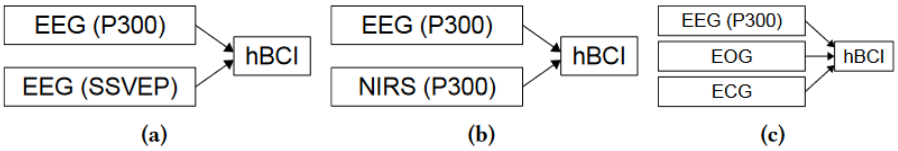


Figure 1.11: There are three conceivable component combinations within hybrid BCI systems. (a) A hybrid BCI system employing two EEG sensors with distinct paradigms. (b) Pure hybrid BCI system employing two BCI devices with the identical paradigm. Mixed hybrid BCI system consisting of one BCI device and two non-BCI devices[5].

1.5.2 Sequential or simultaneous processing

These interfaces can process incoming information sequentially or simultaneously. Sequential processing indicates that the data collected from one component will be utilized prior to the data collected from other components. This is exemplified by the ”brain switch,” in which one component triggers the data processing of another component[76]. This brain switch was employed by Pfurtscheller et al. to regulate an orthosis [8]. Using LEDs and an SSVEP paradigm, the user can select which part of the orthosis to move. The motor imagery paradigm is then employed to move the orthosis effectively. This decreases the number of false positives and exemplifies one application of a ”brain switch”[5]. Simultaneous processing entails processing all the data sent by each component at the same time. This processing method was used by Li et al. to operate a 2D cursor[77]. The P300 paradigm was employed for vertical movement, whereas motor imagery was utilized for horizontal movement. Both components engaged with the system simultaneously[5].

1.5.3 Most prevalent input devices utilized in hybrid BCI

Brain-computer interfaces are frequently utilized in conjunction with a variety of technologies. The section that follows describes these devices and includes examples of prior work.

Eye trackers

Electrooculography, the first type of eye tracker, detects the continuous electric potential field emitted by the eyes as dipoles. The cornea would represent the positive pole, whereas the retina would represent the negative pole[7]. Eye movement can be evaluated by capturing the electric potential field by putting electrodes that, depending on the position of the retina and cornea, will receive a positive or negative signal amplitude. Figure 1.10 illustrates an electrooculography apparatus. Other eye trackers utilize specialized contact lenses or video cameras[78]. Eye trackers can be used to measure numerous motions and events, such as blinking, saccades, and xations. It has been used in conjunction with a BCI to enable impaired individuals to control robots[79] or to target an element on the screen, such as in a spell checker.

Electrocardiography

Electrocardiography utilizes chest-mounted electrodes to detect cardiac activity[80]. It is used to monitor the pace and regularity of the heartbeat and facilitates the detection of heart injury or anomalies. One use of electrocardiography in hybrid BCI systems is effort measuring [76]. This effort assessment can then be utilized to activate or deactivate a BCI device. This system aids in the prevention of false positives.

Electromyography

Electromyography can be performed on the surface of the skin or within the muscle to measure the electrical activity generated during muscular activity. Electromyography is used to extract information on the intensity and duration of muscle activity[**?**]. The figure below depicts an electromyography device attached to a subject. In many BCIs,

the function of electromyography devices is to measure muscle activity during a motor imagery or motor execution job in order to enhance accuracy.

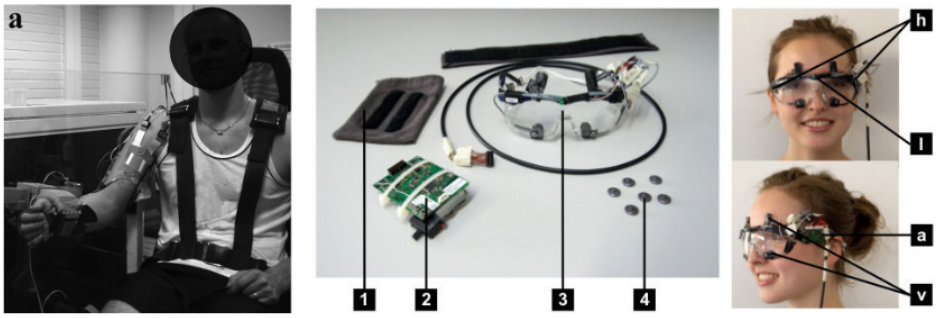


Figure 1.12: a device for electromyography that is placed on a user[0]. (b) Electrooculography device[6][7].

Other input devices

Occasionally, mice, keyboards, and joysticks are also utilized with a BCI. A hybrid system proposed by Leeb et al. consists of a joystick for controlling the movement of a penguin on a slope and a BCI system for triggering jumps [81]. Kreilinger et al. utilized a joystick whose performance degraded over time to represent the user's fatigue [49]. The user was automatically shifted between devices based on a "device score"[5].

1.6 Some instances of hybrid brain-computer connections

This section presents a variety of hybrid BCI setups based on the nature of its component parts: pure hybrid BCIs (consisting just of BCI devices and paradigms) and mixed hybrid BCIs (including non-BCI devices)[5].

1.6.1 Pure hybrid BCIs

Motor imagery and P300 combination

The motor imagery paradigm and the P300 paradigm have been merged in a system created by Riechmann et al.[82]. This configuration suggests a parallel and asyn-

chronous system that incorporates both paradigms. The level of false positives was assessed when subjects were required to use either paradigm. The user was provided with a single target: one of five P300 symbols or one of two motor imagery directions (left or right)[5].

Motor imagery and SSVEP combination

Pfurtscheller et al.[8] have presented a method which integrates motor imaging with SSVEP. Using the SSVEP and motor imagery paradigms, users were able to control a four-step hand orthosis with the aid of this configuration. On the orthosis were mounted two LEDs, one flashing at 8 Hz and the other at 13 Hz. The user utilized these two LEDs to select which portion of the orthosis to open or close. Then, motor imagery was used to complete the job. Each user carried out four distinct tasks. The first permitted the user to open and close the orthosis using only SSVEP. The second task required users to perform rapid foot motions in response to a cue displayed on a computer screen in order to complete the motor imagery paradigm training. The third consisted of using both paradigms to operate the orthosis at one's own pace. The last job consisted of utilizing the SSVEP paradigm exclusively and acted as a control setup. One channel focused on the motor cortex while the other focused on the visual cortex. This hybrid approach demonstrated a lower number of false positives compared to the usage of event-related synchronization continuously[5].

P300 and SSVEP combination

Xu et al.[9] have proposed the combination of the P300 and SSVEP. This configuration sought to improve the performance of a P300 speller by combining SSVEP with P300. Twelve participants conducted an offline spelling test using the two methods. No prior training has been conducted. Six of the patients began with the hybrid configuration, whereas the remaining six began with the P300 paradigm alone. A 3x3 grid was utilized to display the numerals 1 through 9 on a computer screen. In the hybrid system, the highlighted sequence of characters was random. To induce a P300 response, the

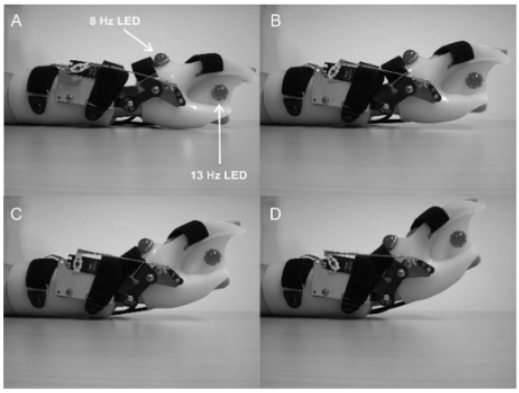


Figure 1.13: A hand orthosis controlled by a BCI. The SSVEP paradigm might be used to determine which component of the orthosis should be activated, whilst the motor imagery paradigm could be utilized to initiate movement execution. (A), (B), (C), and (D) depict four distinct opening procedures for the orthosis[8].

size and font of the characters were altered, and flickering was employed to induce an SSVEP response. Using a hybrid approach significantly increased spelling performance, as shown by the findings[5].

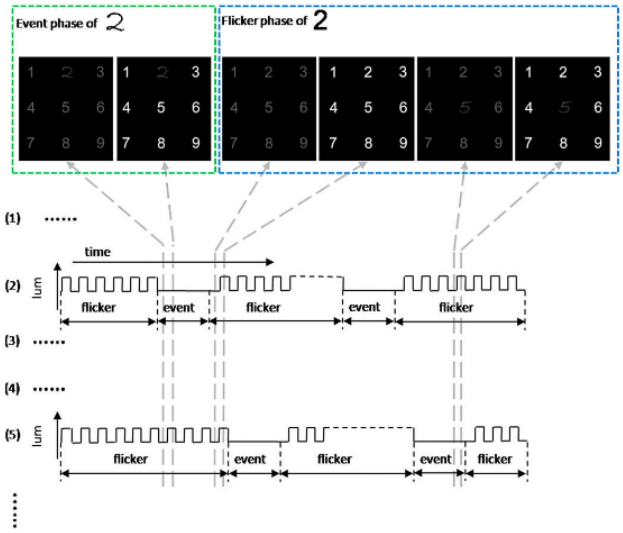


Figure 1.14: Evolution over time of a P300 speller grid utilizing both the P300 and SSVEP paradigms. A ”Event phase” is utilized to stimulate P300 activity, whereas a ”Flicker phase” is used to trigger SSVEP activity.

1.6.2 Mixed hybrid brain-computer interfaces

This section describes a variety of configurations that mix BCI paradigms with non-BCI equipment.

Combining motor imagery with electromyography, eye tracking, or electrocardiography

Leeb et al. integrated EEG and electromyographic data, enabling partially paralyzed individuals to employ residual muscle activity alongside motor execution or motor imagery[83]. In cases of partial paralysis, some residual muscle function can still be utilized. Each trial consisted of xating a cross on a computer screen for three seconds, followed by a five-second signal indicating which hand motor task was to be performed. EEG and electromyography were simultaneously recorded, and their fusion was performed using a naive Bayesian approach with weights that were evenly balanced. This BCI was controlled using either one or all of its components. This configuration demonstrated that the user can maintain control even when fatigued. The classification scores for the EEG component alone were 73%, 87% for electromyography, and 91% for the combination of the two[5]. Eye tracking technologies can be

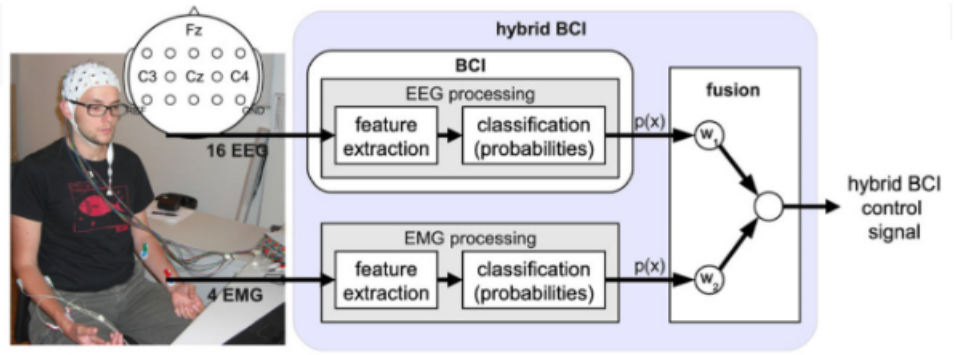


Figure 1.15: A subject with an EEG cap and electromyography electrodes on their head. The graphic depicts the various processing and fusing processes required for each component of the hybrid BCI[9].

utilized to choose certain letters or words. Yong et al.[10] proposed a combination of this device with motor imaging or motor execution. This system selected letters or words using an eye tracking device and motor imagery (or motor execution). The user was required to retain her sight on the desired word or letter for a predetermined amount of time. The system also utilized a ”sleep mode” to limit the number of false positives when no letter or word was shown. Additionally, the classifier used to identify the attempted hand extension was adaptively updated. This arrangement utilized

the application Dynamic Keyboard, which allowed the user to type text by hitting huge buttons and also included a word prediction function.

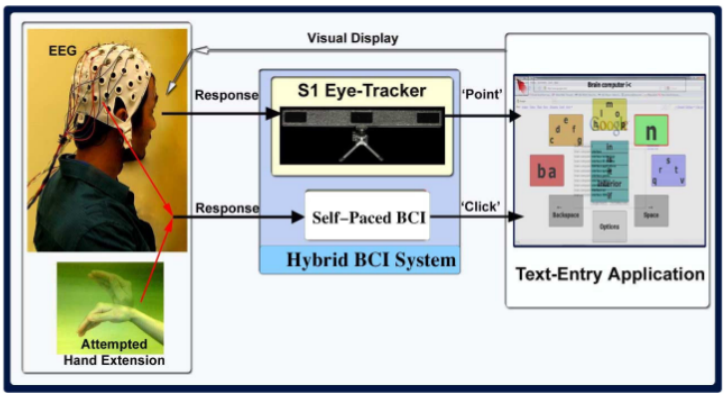


Figure 1.16: A person wearing an EEG cap and electrooculography electrodes, utilizing an eye tracker to indicate a spot on the screen and a hand extension attempt (motor execution or motor imagery, depending on the user's ability to extend her hands) to simulate a mouse click[10].

Using a combination of motor imagery and electrocardiography, Shahid et al. suggested a test to determine if a classification performance could be observed [11]. Multiple trials were conducted, with each lasting 12 seconds: 6 seconds in a calm condition, followed by 6 seconds during which a directional arrow indicated that a left foot or left hand motor imagery task had to be performed. A third symbol represented a state of inactivity. The processing technique is depicted as a block diagram in Figure 1.15. The findings of a classification between a hand motor imagery task and a rest task were subjected to two distinct analyses. The first study, which measured the average heart rate during a motor imagery task, revealed that a 10 percent rise in heart rate may be observed in this situation. The second analysis revealed that the simultaneous use of both devices resulted in an average classification accuracy of 92%, whereas EEG alone resulted in an accuracy of 73%. Due to the reduction of false positives, the combination of electrocardiography and EEG features improved the classification of motor imagery in both training and evaluation.

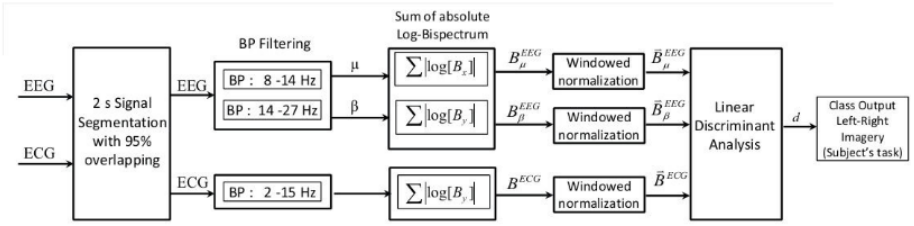


Figure 1.17: A block diagram of a system that combines motor imagery with electrocardiography to evaluate a potential increase in classification ability[11].

Combining P300 and electrooculography

A method devised by Postelnicu et al. used a modified stimulus presentation paradigm called the ”Half Checker Board Paradigm” to improve spelling speed by combining the P300 paradigm and an electrooculography device[12]. Using electrooculography, an 8x9 matrix was subdivided into chosen sections. Multiple characters and symbols were highlighted at random in each of these regions. EEG was utilized to choose the characters. Every two characters entered required a phase of electrooculography calibration. The users were required to compose a 13-character calibration text and a 16-character evaluation paragraph. When employing the half checkerboard paradigm, the time required to spell one character decreased, resulting in a more efficient P300 paradigm.

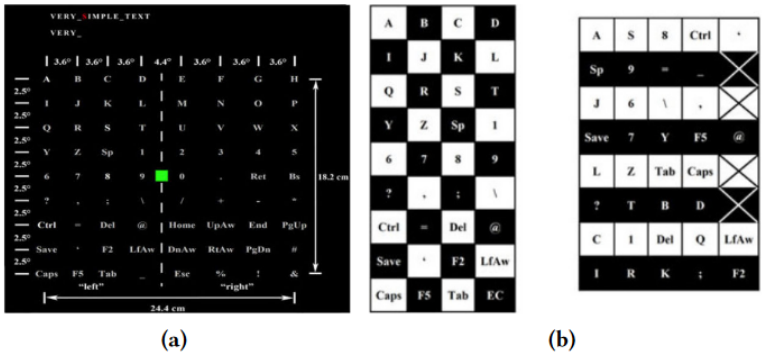


Figure 1.18: A character matrix for spelling purposes. The matrix (a) is subdivided into selectable sections (b) using an electrooculography device, while character selection is conducted using the P300 paradigm[12].

1.7 Related Works

Deep Learning (DL) has recently gained significant interest from a variety of disciplines due to its promising performance in categorization problems[84]. Numerous studies have demonstrated that deep learning will play a crucial role in the exact decoding of brain processes. Recent research has focused specifically on EEG motion intention detection. Implementing the DL-based method for decoding EEG MI tasks, which has yielded encouraging results, is a current priority.

1.7.1 Control of a wheelchair by motor imagery in real time

An implementation of a non-invasive brain machine interface (BMI) for operating a motorized wheelchair. Subjects were educated utilizing an effective feedback training method, after which they were able to freely maneuver the wheelchair, similar to using a joystick. Through the use of efficient signal processing techniques and a feedback training method, the subjects' training time and effort have been lowered while their accuracy has been improved. Using the motor imagery paradigm, trained subjects could freely maneuver a wheelchair with several degrees of freedom and required rapid response time. The used classifier in this experiment was Support Vector Machine (SVM) and the output was turning right or turning left[85].

1.7.2 Quad-copter control in three-dimensional space

Using noninvasive scalp electroencephalography (EEG) to control a robotic quadcopter in three-dimensional (3D) physical space with human beings. The authors then measure the performance of this system utilizing metrics appropriate for asynchronous BCI. Lastly, they compare the effect of operating a real-world gadget to a 2D virtual cursor task on the control of individuals. Five human individuals were trained to modify their sensori-motor rhythms in order to navigate a 3D physical world while controlling an AR Drone. On the drone's hull, a forward-facing camera offered visual feedback. Key results. Individuals traveling at an average straight-line speed of 0.69 m s-1 were able to accurately acquire up to 90.5% of all valid targets given. Signifi-

cance. The ability to freely explore and engage with the surrounding environment is a critical aspect of autonomy that is lost in the context of neuro-degenerative disease. Brain-computer interfaces are systems that try to restore or improve a user's capacity to interact with the environment using only mind and a computer. They demonstrate for the first time that human EEG captured from the scalp may be used to operate a flying robot in 3D physical space. Our research demonstrates the potential for noninvasive EEG-based BCI systems to provide intricate spatial control in three dimensions. Using tele-presence robotics, the present study may serve as a platform for the examination of multidimensional noninvasive BCI control in a physical context. A linear classifier has been used to classify 6 different movements[86].

1.7.3 EEG based BCI for controlling a robot arm movement

A unique BCI system that utilizes the user's thoughts to operate a robot arm. Four participants aged between 20 and 29 (one female and three males) participated in our investigation. They have been told to visualize the execution of actions of the right hand, the left hand, both the right and left hands, or the feet, depending on the set protocol. Using an EMOTIV EPOC headset, neuronal electrical impulses from the subject's scalp were recorded and transferred to a computer for analysis. Utilizing the Principal Component Analysis (PCA) technique in conjunction with the Fast Fourier Transform (FFT) spectrum within the frequency region important for sensorimotor rhythms (8 Hz–22 Hz), feature extraction was carried out. The outputs of a Support Vector Machine (SVM) classifier based on a Radial Base Function (RBF) were then translated into commands for controlling the robot arm. The suggested BCI enables the control of the robot arm in four directions, including right, left, up, and down, with an average accuracy of 85.45% across all individuals[87].

1.7.4 Control of Humanoid-Robot using an EEG-based BCI

An innovative interface for translating human intents into motion orders for robotic devices. The experimental approaches include offline training, online testing with

feedback, and real-time control sessions. EEG amplitude features are retrieved using power spectral analysis, and informative feature components are chosen using the Fisher ratio. The two classifiers are trained to identify human intentions and are structured hierarchically to construct an asynchronous BCI system. For the performance test, five healthy volunteers navigated a humanoid robot through an indoor maze using their EEGs in conjunction with real-time pictures obtained from the robot's head camera. Using the suggested asynchronous EEG-based active BCI system, the subjects were able to successfully control the humanoid robot in the indoor maze and achieve the intended destination[88].

1.7.5 Classification of motor imagery using deep convolutional neural networks

Deep convolutional neural network (CNN)-based approach for feature extraction and categorization of single-trial MI EEG. First, on the basis of the spatio-temporal characteristics of EEG, a 5-layer CNN model is constructed to classify MI tasks (left hand and right hand movement); next, the CNN model is applied to the experimental data set collected from subjects and compared to three conventional classification methods (power + SVM, CSP + SVM, and AR + SVM). The results reveal that CNN may further improve classification performance: the average accuracy using CNN (86.41 0.77%) is 9.24 percentage points, 3.80 percentage points, and 5.16 percentage points higher than those obtained using power + SVM, CSP + SVM, and AR + SVM, respectively. The proposed method was effective for classifying MI and provides a realistic method for classifying EEG signals non-invasively in BCI applications[89].

1.7.6 LSTM-Based EEG Classification

A classification system for EEG data in motor imagery challenges. The suggested framework consists of multiple steps and components, namely preprocessing, feature extraction utilizing the 1d-AX, channel weighting, LSTM network, and softmax regression. In contrast to the majority of previous methods that employ CSP techniques

and deep networks, LSTM networks are used to extract critical features of EEG data so that time-varying EEG signal characteristics can be exploited to improve classification performance. In addition, the combination of 1d-AX with channel weighting makes the EEG signal representation more concise, which facilitates the subsequent training of LSTM networks. The experimental results based on the public BCI competition dataset reveal that the proposed AX-LSTM has superior classification performance. For online classification, the entire network can be trained offline using EEG data from many subjects. It is then fed additional EEG segments and outputs the prediction label probability. The scale of parameters in the proposed method is significantly less than in existing deep neural networks, resulting in faster real-time processing and a reduced likelihood of over-fitting[90].

1.7.7 Deep learning CNN-WNN approach to classifity EEG based BCI

A unique deep learning algorithm with data augmentation to increase the classification accuracy of motor imaging signals and prevent overfitting. Complex Morlet wavelets were used to turn EEG data into tensors as inputs to the neural network. Two models, Convolutional Neural Networks and Wavelets Neural Networks, were presented to identify and extract features from motor imaging inputs. To train the networks, the empirical mode decomposition method was used to produce artificial EEG frames. In the experiments, we examined the filter size and hyper-parameters, as well as the epoch size for both networks. The optimal batch size was determined to be 300. In order to test the performance of the suggested method, CNN and WNN models were compared to tensor decomposition techniques. The experimental findings demonstrate that our innovative approaches are superior to the tensor decomposition method. In addition, we validate our suggested approaches using dataset III from the BCI Competition II. The accuracy of the competition's winning algorithm was 89.3%, whereas the accuracy performance of our proposed approaches was 90.1%. Specifically for the WNN model, the bigger artificially created EEG frames can result in the network display-

ing an exception of no convergence. In order to evaluate the viability of the newly suggested WNN model, the actual motor imagery dataset was used to eliminate the influence of the false EEG frames. According to the results, the WNN model has greater classification accuracy and a faster convergence rate than the CNN model[91].

1.7.8 Attention-based BiLSTM-GCN for Human Motor Imagery Recognition

A revolutionary deep learning system aimed to recognize motor imagery (MI) with exceptional accuracy and responsiveness using scalp EEG. Bidirectional Long Short-term Memory (BiLSTM) with the Attention mechanism is employednvolutional neural network (GCN) improves decoding performance by collaborating with the predicted topological structure of features. Particularly, this method is trained and tested on short EEG recordings with a duration of only 0.4 seconds, and the outcome has demonstrated effective and efficient prediction based on individual and group-wise training, with 98.81 percent and 94.64 percent accuracy, respectively, outperforming all previous studies. Deep feature mining can perfectly distinguish human motion intents from raw and practically immediate EEG signals, paving the way for EEG-based MI recognition to be implemented in a viable BCI system[92].

1.8 Problematic

Drugs, alcohol, Parkinson's, and other things have a big effect on brain cells, so they have a big effect on the quality of an EEG. Most of the research only looks at EEGs from people who are healthy in order to improve how they are used. One of those implementations is brain-controlling (games, robots, etc). However, this is not an easy task to generalize among people, and that's because of the following problems: (i) EEG waves are not stable enough to deal with and this belongs to the human profile and addictions (especially alcohol and drugs). (ii) Abnormal EEG are hard to classify. Thus, the generalization task for allowing different types of subjects to control robots using their minds is still hard, especially for Parkinson's patients, drug addicts, and

alcoholic people. (iii) Decoding of motor imagery is fast with machine learning but not accurate enough with deep learning. (iv) Most current work for brain-controlled robots is not in real time (some models are slow). (v) It's very hard to control more than a robot at once (multi-robot controlling issue).

1.9 The objective of our study

In this study, we proposed a novel multi-method approach that uses simultaneous models (Deep/Machine Learning) in order to facilitate the classification of abnormal EEGs and generalize the mind-controlling task for several subjects with different backgrounds (healthy, drugged, and alcoholic). Second, we try to make the mind-controlling task executed in real-time by speeding up our model's predictions. Lastly, we aim to realize the multi-robot controlling task, which allows the patients to control several different robots (UAVs, wheeled mobile robots,... etc).

1.10 Multi-Method Approach

As discussed previously, we used 2 simultaneous models in order to increase the precision of our system. The multi-method approach is an auction-based system that uses two different simultaneous models to enhance the prediction. In our study, we used different combinations; SVM and Bi-LSTM attention-based, Bi-LSTM with GCN, etc. The multi-method approach works as follows: we compute the average result over a specified time period range. The range depends on the execution time of the model. Since the previous results were done in real time, we guessed that the period range would be 5Hz. This means that the multi-method approach will figure out the average precision over 5 trials, and then the controller will use the most accurate prediction to move the robots. The major reason for using this method is to reduce the sensitivity of the robots, so the execution time for each move will be 1 s (imagine the sensitivity of the robots if the user used different maneuvers simultaneously). The benefit of this method is that it makes robots easier to control by decoding both linear and nonlinear features that come from an abnormal EEG (a sign of poor health).

1.11 Conclusion

This chapter provides an overview of hybrid brain-computer interfaces, including how its BCI components can be classified, merged based on their nature, and processed in real time (sequentially or in parallel). Some examples of regularly used input devices inside a hybrid BCI and typical examples of hybrid BCI systems have been provided. Then we highlighted some previous works, highlighting their methods, techniques, and outcomes and we provide the common used BCI applications. Those works are connected in some way, particularly with ours. Lastly, we talked about the problems EEG is having with controlling robots and what we want to do to solve them.

Chapter 2

Technical Background

2.1 Introduction

This chapter discusses the background material pertinent to the design and implementation of a brain-computer interface, including a review of the various recording and signal processing techniques required to produce a usable signal, and a discussion of BCI classification techniques. The following diagram provides an overview of these steps, which include[2]:

1. **Brain Signal Acquisition:** Recording neural data
2. **Preprocessing:** Utilizing signal processing techniques, such as filtering and artifact removal, to enhance signal quality.
3. **Feature Extraction:** Identification and extraction of informative elements from brain data recordings
4. **Classification:** Using the retrieved information to develop a classifier capable of distinguishing between various cognitive states
5. **Device Control:** Using the outputs of the classifier as a communication channel that can communicate directly with an external device.
6. **Feedback:** Providing the user with real-time feedback that enables them to adjust their strategy as needed to increase performance.

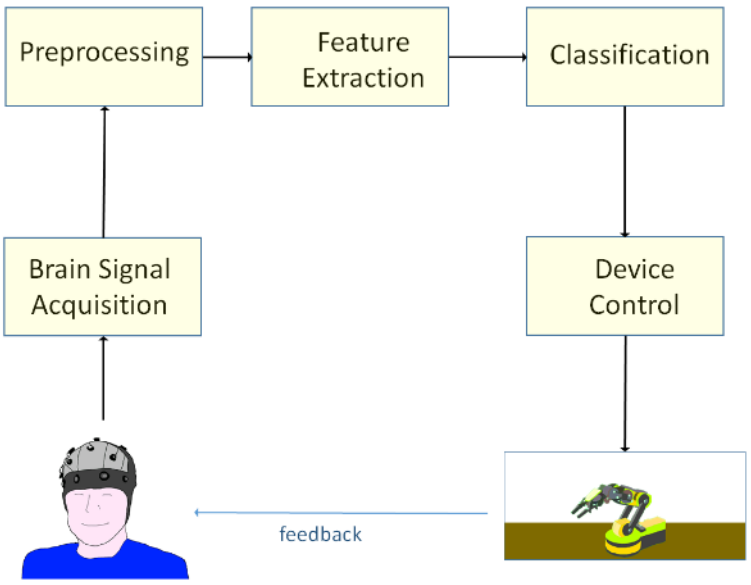


Figure 2.1: BCI framework[2]

2.2 Brain Signal Acquisition

Methods for recording brain activity can be categorized into two major categories: invasive methods, which require insertion of some components into the body, and non-invasive methods. This section describes the most popular available approaches[2].

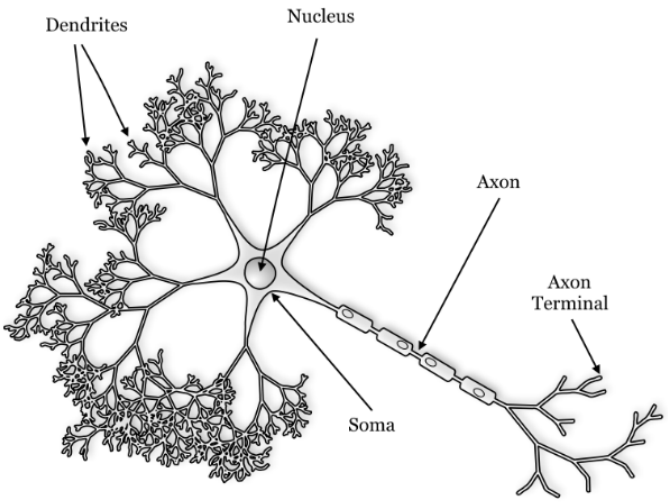


Figure 2.2: Neuron diagram by Nicolas Rougier (licensed under CC BY 3.0)

2.2.1 Invasive Methods

Implanted arrays of intracortical microelectrodes in the cerebral cortex of the brain provide a resolution that is both spatial and temporal. Collinger et al. [93] and Hochberg et al. [94] proved that tetraplegic users could learn to manipulate a robotic arm with 7 degrees of freedom. The main disadvantages of intra-cortical electrodes are that they need surgery and that the brain can recognize them as a foreign body, frequently resulting in inflammatory reactions such as ”glial scarring” [95]. Electrocorticography (ECoG) is the measurement of cortical field potentials utilizing electrodes inserted on the cerebral cortex's outer surface. As it is placed below the skull but outside the brain, technically speaking, ECoG might be classified as somewhat invasive. ECoG is advantageous because to its great temporal and spatial resolution. The primary problem is that electrode implantation requires surgery.

Figure 2.3: Dry EEG headset

2.2.2 Non-invasive Methods

Electroencephalography (EEG), the primary technique utilized for this thesis, monitors variations in electrical potential induced by neuronal activity in the brain. Electrodes placed on the scalp are utilized to acquire EEG signals. Signal extraction involves the coordinated activation of thousands or millions of cortical neurons [46], mostly because EEG scalp potentials become distorted as they move through the brain, skull, and scalp tissues. Typically, scalp electrodes are positioned in accordance with the

10-20 system, where electrode spacing between consecutive electrodes is either 10 or 20 percent of the diameter of the skull, front-to-back or left-to-right[96]. The labels Fp, F, C, P, and O denote the frontopolar, frontal, central, parietal, and occipital brain areas, respectively, when referring to electrode sites. Electrodes for electroencephalography (EEG) might be water-based, gel-based, or dry (Fig. 2.4). Gel electrodes can be time-consuming to install and require gel to be applied to the hair of participants, which might be uncomfortable for certain users. It has been discovered that dry electrodes contribute to an increase in impedance as well as broad-band noise[97], which negatively impacts signal quality. Magnetoencephalography (MEG) is a technique for measuring the magnetic perturbation induced by neural activity. Over the scalp, an array of superconducting quantum interference devices (SQUIDs [98]) is used to capture MEG data. Functional magnetic resonance imaging (fMRI) identifies alterations in blood oxygenation level; these alterations are related to brain activity. fMRI is performed in an fMRI machine, which generates a magnetic field that modifies the status of protons within the body. Different rates of'relaxation' (the time required for protons to return to their initial state) enable researchers to calculate the blood oxygenation level in every region of the brain at a given time. NIRS is also utilized to measure brain changes related to blood oxygenation levels. On the surface of the head, near-infrared light can pass through skin and bone and is absorbed by haemoglobin. Researchers can monitor the oxygenation and haemodynamic activity associated with brain activity by measuring the absolute change in oxyhaemoglobin (HbO2) and deoxyhaemoglobin (Hb) by measuring the absolute change in oxyhaemoglobin (HbO2) and deoxyhaemoglobin (Hb)[99]. The aforementioned methods vary in terms of their respective strengths and weaknesses: fMRI has poor temporal resolution (1-2 seconds) and good spatial resolution (order: 64mm3) [100]. The spatial resolution of near- infrared spectroscopy (NIRS) techniques is on the order of centimeters, but the depth resolution is inadequate [101].

2.2.3 EEG Headsets

The number of portable, low-cost EEG-based systems on the market has expanded in recent years[102]. Continuous recording of EEG data and/or the reproduction of bigger EEG analytical systems using portable devices have been the focus of research on the usage of low-cost EEG systems. In this review, we examined research papers that detailed the use of low-cost EEG devices, with an emphasis on devices where the headset cost less than $1,000 USD, excluding license fees: the InteraXon Muse, the Neurosky MindWave, the Emotiv Epoc, the Emotiv Insight, and the OpenBCI. These gadgets are representative of widely distributed commercial models. Although other devices and suppliers have been used [103], the focus of this search was on non-invasive EEG devices that were less than $1000, not marketed as medical devices, accessible to consumers, prominent in the hobbyist community, and provided tools or options for brain-computer interface (BCI) applications. Table 1 compares several commercially available, low-cost EEG headsets. The majority of inexpensive headphones feature dry electrodes, which are more convenient for casual users. Similarly, the majority of headsets incorporate research tools, open-source software, and additional hardware as packaged software[104][105]).

Device	Electrodes	Sampling Rate	External Information
InteraXon	-Rigid electrode placement	- 256 Hz	- Research Tools for Windows, Mac, and Linux
Muse v1, v2	- 4 channels: AF7, AF8, TP9, TP10	- 12 bits	- Source Developer Kit (SDK) for Android, IOS, Windows
			- Cost: $200 USD
Neurosky MindWave	- Rigid electrode placement	- 512 Hz	-SDK Available
	- 1 channel: AFz	- 12 bits	- Cost: $99.99 USD
OpenBCI	- Up to 16 channels	- 256 Hz	-Open-source software, firmware, and hardware
	- Flexible electrode placement at 35 locations	- 24 bits	-Cost: $500 USD for 8 channels, $949 USD for 16
Emotiv Epoc, Flex, and Insight	- Rigid electrode placement	- 128 Hz	-Research Tools for Windows, Mac, and Linux
	- Epoc: 14 channels (AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4) -Insight: 5 channels (AF3, AF4, T7, T8, Pz)	- 14 bits	-Cost: $799 USD (Epoc), $299 USD (Insight)

Figure 2.4: Comparison of consumer EEG headsets[13].

InterAxon Muse

The InteraXon Muse is a tiny EEG system that detects brain activity using four EEG sensors (Muse, InteraXon) and can transfer data to neighboring devices via Bluetooth. Muse claimed that the headband might facilitate a state of profound relaxation. The dry electrodes were positioned at FPz, AF7, AF8, TP9, and TP10 according to 10–20 International electrode placement convention[106]. The reference electrode was the FPz electrode. The detailed specifications match those of the original Muse gadget.

Neurosky Mindwave

Neurosky designed the MindWave as a low-cost, single-channel, dry EEG headset that can transmit EEG wirelessly over Bluetooth Low Energy or traditional Bluetooth [107]. The MindWave device is a headset with a T-shaped headband, a larger ear clip, and a bendable arm. The reference and ground electrodes of the device are placed on the ear, whilst the EEG electrode is put on the forehead, above the eye. Neurosky EEG headphones provide training software, instructional programs, and information for software developers. For researchers, there are also available free development tools. Despite the fact that Neurosky produces several models, the MindWave was the most commonly utilized model in the studies considered[104][107]).

OpenBCI

The OpenBCI Ultracortex Mark IV is a 3D-printable, open-source headset designed to function with any OpenBCI board. It is capable of recording EEG of a research-grade quality. Based on the 10–20 International System, the Ultracortex Mark IV headset is capable of sampling up to 16 channels of EEG from up to 35 different places[108]. The OpenBCI boards are available with four, eight, or sixteen channels. The OpenBCI is a collection of open-source components that must be assembled prior to use[109]. Therefore, it is not as commonly utilized as consumer gadgets easily available for purchase, but it theoretically allows for greater flexibility. It has previously been utilized to identify tiredness in a driving simulator.

Emotiv Insight and Epoc

Emotiv offers both the smaller, less priced Insight and the larger, more expensive Epoc (as well as the improved Epoc+). The Emotiv Epoc is the most expensive of the researched EEG headsets due to its greater number of electrodes [110]. Each of its two electrode arms contains two sensor and two reference electrodes. These areas cover the temporal, parietal, and occipital lobes. Emotiv offers a free companion application for monitoring emotions. In addition, they provide pay-to-download games such as Arena, which enables users to experience mental commands. Emotiv provides an SDK with two tiers for the Epoc. The headset has been utilized in scientific research, including BCI and brain state detection[111][112][113]. However, the Epoc and Epoc+ models were the most prevalent. Using "Insight" as a keyword resulted in references to the Epoc and Epoc+.

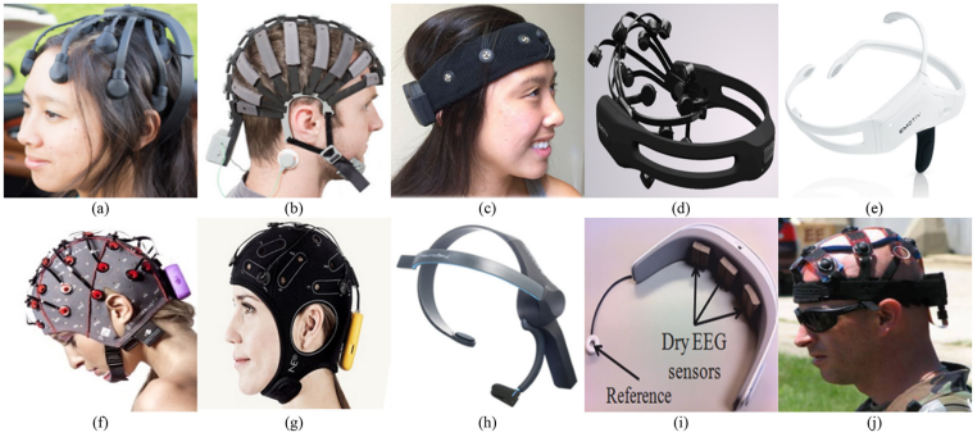


Figure 2.5: Diverse wireless portable headsets and acquisition methods for EEG recording. (a–c) Miniature Wireless Acquisition Systems by Cognionics with the Quick-20 Dry EEG Headset, 72-Channel Dry EEG Headset, and Multiposition Dry EEG Headband, respectively; (d and e) EPOC and Insight wireless EEG acquisition systems by Emotiv; (f) g.Nautilus wireless EEG acquisition system by g.tec; (g) ENOBIO 8 wireless EEG system by Neuroelectrics; (h) MindWave Double-column image proportions.

2.3 Preprocessing

The objective of preprocessing is to enhance the signal-to-noise ratio (SNR) and spatial resolution by eliminating artifacts. Artifacts are undesirable additions to the signal

that might contribute positively or negatively to the performance of a BCI. Artifacts can be removed by referencing and filtering the data, among other techniques. Sanei and Chambers[114] identify a variety of potential artifact sources, including muscles (electromyogram, EMG), eyes (electrooculogram, EOG), electrical interference, and cable defect artifacts. Signal amplification and filtering are necessary to remove artifacts. The common EEG artifacts are listed previously. In thresholding, a threshold is set for an input signal (such as EOG), and epochs where the signal's amplitude exceeds the threshold are considered tainted and deleted[2].

2.3.1 Downsampling

EEG data is incredibly high-dimensional due to the intricacy of brain activity, making it naturally challenging to classify. Downsampling is a dimension reduction technique that decreases the sampling rate. By removing every other sample, EEG recordings at 1000 Hz can be downsampled to 500 Hz. This minimizes complexity and, if utilized correctly, can increase BCI performance[2].

2.3.2 Temporal Filtering

Discrete Fourier Transform Filters The Discrete Fourier Transform (DFT) is a method for transforming a signal from the time domain to the frequency domain by eliminating all temporal information and expressing the signal as a sum of sinusoids. The DFT filtering of a signal x_n requires three steps: translating the signal into the frequency domain, setting coefficients outside of the desired range to zero, and finally transforming the signal back into the time domain[2]. DFT filtering can be accomplished by:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} kn} \tag{2.1}$$

k is the sinusoidal frequency at k/N samples, e is Euler's constant, and i is an imaginary number with i^2 = -1, and N is the number of samples[2]. The signal is converted back to the temporal domain using the inverse DFT (IDFT) after all coefficients outside of

the desired frequencies are set to zero:

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{\frac{i2\pi kn}{N}} \tag{2.2}$$

Finite Impulse Response Filters A finite impulse response (FIR) filter is a linear filter with a limited-length input response. The FIR response is computed using the previous M unfiltered signal samples $s(n)$. Find the filtered signal $y(n)$ using:

$$y(n) = \sum_{k=0}^{M} b_k s(n-k) \tag{2.3}$$

where b_k is a vector of feed-forward filter coefficients and $s(n)$ is the unfiltered, unprocessed signal.

Infinite Impulse Response Filters Infinite impulse response (IIR) filters are recursive digital filters with an infinitely long input response[2]. The IIR response is dependent on both the last M samples of $s(n)$ and the prior P filter operations' outputs. Find the filtered signal $y(n)$ using:

$$y(n) = \sum_{k=0}^{M} b_k s(n-k) + \sum_{k=1}^{P} a_x y(n-k) \tag{2.4}$$

where a_x is a vector holding the coefficients of the feedback filter.

2.3.3 Temporal Filter Applications

1. **High-pass filtering:** Low-frequency signals are frequently accompanied by artifacts, such as those caused by breathing, amplifier drift, and variations in skin resistance caused by perspiration. The majority can be eliminated with a high-pass filter with a cutoff frequency between 0.5 and 1 Hz. EEG may also detect electrocardiogram (ECG) artifacts[115], but the impacts of this low-frequency signal can be mitigated by a high-pass filter.

2. **Low-pass filtering:** Typically, low-pass filters with a cutoff frequency of 50-70

Hz are used to eliminate high-frequency noise[114].

3. **Band-pass filtering:** Using band-pass filters, several relevant frequency bands can be extracted, including those connected with motor imagery, such as the mu and beta bands. Even BCIs that do not rely on spectral data, such as the P300-BCI, are typically filtered before to detection. Typically, P300-BCI signal identification requires band-pass filtering between 0.1 and 20 Hz.

4. **Notch filtering:** Notch filters are often extremely high-order band-stop filters. They can eliminate 50 or 60 Hz line noise.

5. **Zero-phase filtering:** In order to prevent phase distortions and signal delay, zero-phase filters apply a time reversal to data during the filtering process. The filter operates by first filtering the data, then reversing and filtering the data again, and then reversing the data once again. Zero-phase filtering is often reserved for offline data due to the fact that it is non-causal and depends on future inputs.

2.3.4 Spatial Filtering

Reference Electrodes In EEG BCIs, reference electrodes are utilized to determine the channel voltage. The voltage, which is the difference in electrical potential between two places, is determined by placing the reference electrode in close proximity and computing the difference. The most common reference site is the mastoid bone (behind the ear)[116][117][118]; nevertheless, tactical reference placement might give considerable advantages.

Scalp Reference Using a scalp electrode as a reference eliminates common noise in that region of the brain. Numerous investigations, such as SSVEP studies utilizing references in central and parietal regions[119][120][121], have utilized this technique to isolate SSVEP activity, which is typically best detected by electrodes above the occipital lobe.

Bipolar Reference By subtracting s_j , the signal from channel j, from s_i, the signal from channel i a new bipolar channel $\tilde{s}_{i,j}$ is created.

$$\tilde{s}_{i,j} = s_i - s_j \tag{2.5}$$

Common Average Reference Common average reference (CAR) operates by subtracting, at each time point, the average signal of all electrodes from the signal of each electrode. This approach is effective in reducing noise common to all electrodes (e.g., 50 or 60 Hz power supply noise) and boosting signals contained in a small number of electrodes. However, it is not effective in reducing noise that is shared by a small number of electrodes, such as EOG or EMG. Because of this, CAR is typically employed in concert with other techniques to remove additional artifacts. Utilized when CAR is applied on an electrode montage with N electrodes:

$$\tilde{s}_i = s_i - \frac{1}{N}\sum_{i=1}^{N} s_i \tag{2.6}$$

where N is the number of channels and $\tilde{S}_i$ is a single channel with spatial filtering.

Surface Laplacian Adjusting the signal at each electrode by subtracting the average of the four adjacent electrodes ('small Laplacian')[122] or the four adjacent electrodes ('large Laplacian')[123] is how Laplacian reference works. This technique is helpful for decreasing regionally-specific noise.

$$\tilde{s}_i = s_i - \frac{1}{4}\sum_{i\in\Theta} s_i \tag{2.7}$$

Θ stands for the electrodes of the small or large Laplacian reference. Individually or in some situations jointly, a variety of referencing techniques may be utilized. The objective is to employ approaches that remove the greatest noise from the signals of interest without sacrificing too much meaningful information.

Common Spatial Pattern (CSP) CSP is a technique that identifies spatial filters that maximize the variance between two conditions' EEG signals. Ramoser et al.[124] popularized its application in BCI research, and since then it has undergone various adjustments[125] and expansions for multiclass categorization[126]. Common Spatial Pattern (CSP) functions by locating spatial filters w that maximize variance in one class and minimize variance in another. A fully trained CSP spatial filter formats the data such that the top row's activity correlates predominantly to one class and the bottom row's activity predominantly to the other class. CSP is especially successful in BCIs dependent on oscillatory activity, such as separating left- and right-hand motor imagery. Initial bandpass filtering of data into a suitable band, such as 8-30 Hz (which includes both the alpha and beta rhythms). Next, the normalized spatial covariance of the EEG is calculated by:

$$C = \frac{EE^T}{\text{trace}\,(EE^T)} \tag{2.8}$$

where E is the bandpass-filtered EEG data of size NxT, N is the number of channels, and T is the sample size. Trace $(\cdot)^T$ is the sum of the diagonal components of a square matrix, while $trace(\cdot)$ signifies the transposition operator. Next, the spatial covariance $\bar{C}_d, \in [l,r]$ is derived by averaging the values of C for each class across all trials. Therefore, the composite spatial covariance is:

$$C_c = \bar{C}_l + \bar{C}_r \tag{2.9}$$

C can be expressed in terms of eigenvalues and eigenvectors:

$$C_c = U_c \lambda_c U_c^T \tag{2.10}$$

where λ_c is the diagonal matrix of eigenvalues and U_c is the matrix of eigenvectors. Next, deviations within U_c are equalized by means of the whitening transform:

$$P = \sqrt{\lambda_c^{-1} U_c^T} \tag{2.11}$$

By transforming $\bar{C}_l$ and $\bar{C}_r$, so that:

$$S_l = P\bar{C}_l P^T \quad \text{and} \quad S_r = P\bar{C}_r P^T \tag{2.12}$$

It can be demonstrated that S_l and S_r share eigenvectors. If:

$$S_l = B\lambda_l B^T, \quad \text{then } S_r = B\lambda_r B^T, \quad \text{and} \quad \lambda_l + \lambda_r = I \tag{2.13}$$

where I denotes the identity matrix and B the eigenvectors. Due to the fact that two corresponding eigenvalues total to one, an eigenvector with a large eigenvalue for S_l will have a small eigenvalue for S_l, which provides the CSP method its ability to successfully segregate the variances between classes. Lastly, the projection matrix $W = (B^T P)^T$ is utilized for spatial filtering:

$$Z = WE \tag{2.14}$$

where Z represents the spatially filtered trial. The last chapter describes in detail how this method can be used to extract class-relevant features from EEG.

2.3.5 Source Localisation

The activity detected by EEG sensors placed over a particular brain region is not necessarily indicative of the activity occurring at that region. In addition to passing through layers of bone, skin, and hair, EEG electrodes cover a considerable region. Source Localisation (SL) is a source reconstruction method that models the spatiotemporal dynamics of the brain's neuronal currents using multichannel EEG data. SL operates by translating EEG onto a source grid with a higher dimension[127], where dipoles indicate individual source activity. SL typically requires an MRI imaging of the user's head, from which an anatomical model can be constructed; however, if this is not possible, a standard model can be altered to match the user's head. Forward modeling seeks to recreate EEG data given the source activity; inverse modeling aims to estimate the

current source locations and intensities that produce a given collection of EEG data.

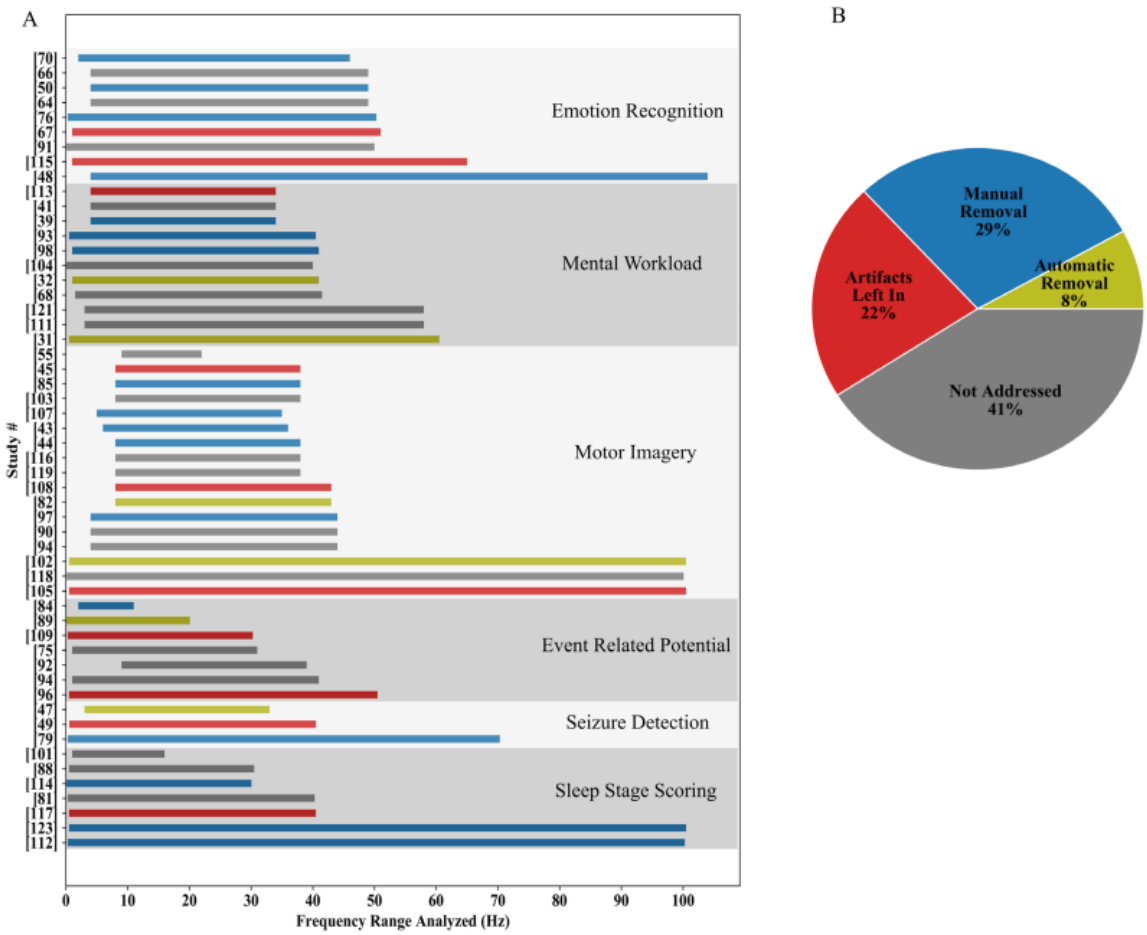


Figure 2.6: Strategies for artifact removal and filtering (A) Frequency range utilized for EEG analysis in each identified investigation, arranged by task type. The bar colors correspond to various artifact removal tactics. Red bars reflect research that addressed the deliberate decision to leave artifacts as data pollutants, while dark grey bars represent studies that did not address any artifact removal approach. The studies are categorized by application type. (B) The proportion of distinct artifact removal procedures utilized in all investigations.

2.4 Artifacts removal and Preprocessing methods

In this section we will discuss the popular used methods for artifacts removing and EEG preprocessing.

2.4.1 Regression Methods

The regression method[128] is a standard way to artifact reduction; it eliminates estimated artifacts by finding the amplitude relation of the reference. Therefore, elec-

trocardiogram (ECG) or electrooculogram (EOG) signals are necessary to distinguish artifacts from EEG signals. The dependence of this approach on reference channels for ECG and EOG elimination is a limitation[129], despite the fact that the regression method is based on simple mathematical understanding and is consequently commonly utilized due to the low computation required.

2.4.2 Blind Source Separation Methods

Blind source separation (BSS) approaches are based on the idea that the signals observed on a multi-channel recording are the result of the combination of several different originating signals; hence, neither additional reference channels nor previous information is required[130]. Principal component analysis (PCA), independent component analysis (ICA), and canonical correlation analysis are typical applications of the BSS technique (CCA). ICA[131][**?**] is a statistical approach used to solve the BSS problem that takes random factors into account to optimize the output components' independence through the finding of a linear transformation. In addition, ICA is a powerful approach that minimizes signal dimensionality and extracts independent components. This approach is effective in extracting artifacts such as eyeblinks and heartbeats because they are produced by independent sources and are not associated with specific frequencies. Under the premise that the original signals are statistically independent and have a non-Gaussian distribution, ICA is efficient. In addition, the signal dimension must exceed that of the source signal. PCA[132][133] optimizes the variance of transformed data based solely on the second-order statistics of covariance[130]. PCA is an established technique for reducing the dimensionality of features while preserving their statistical information. This strategy has the advantage of preserving the variance of the data set. However, if the potentials of drifts and EEG data are comparable, PCA is unable to identify the required interferences. CCA is commonly utilized in SSVEP-based brain-computer interfaces (BCIs) to determine the frequency components of EEG that characterize visual stimulus frequencies[134]. Notably, a comparison of BSS approaches is required, but that's outside the scope of this review article.

The reference[135] provides additional details regarding these strategies.

2.4.3 Wavelet Transform

Wavelet transform (WT)[136][137][138][139] is a spectral estimating technique that turns a time-domain signal into a signal in the time and frequency domains. After decomposition of wavelet transformation on EEG data and during artifact removal, WT localizes the features and preserves them during the filtering process by defining a noisy signal elimination threshold. While WT performs well when examining the components of non-stationary signals, it is unable to identify artifacts that overlap with spectral features. Consequently, novel hybrid approaches such as wavelet-BSS have been proposed[140] to counteract this shortcoming.

2.4.4 Filtering Methods

Frequency filtering, adaptive filtering, and Wiener filtering[141] are among the various filtering methods that have been utilized for EEG artifacts and noise cancellation.

Adaptive filtering: The assumption underlying adaptive filtering is that the EEG signal of interest and the artifact are uncorrelated. This filter employs a reference signal to provide an estimated signal that is correlated with the artifact; the estimate is then subtracted from the source signals to obtain a noise-free EEG signal[142]. Adaptive filtering use the least mean squares (LMS) technique, which converges in a linear fashion, to evaluate the clean signals by enhancing the weight parameter. Another optimization algorithm, the recursive least squares algorithm, is an extension of LMS[129] with quadratic convergence. Depending on the type of recursive least squares method, its convergence may be faster than that of the LMS algorithm, but it has a higher computing cost. A downside of adaptive filtering is that it requires additional sensors to provide reference input[142].

Wiener Filtering: Wiener filtering is a statistical technique that generates a linear time invariant filter to minimize the mean square error between the signals of interest

and the estimated signals[141]. Although Wiener filtering does not require an additional reference, the minimization approach used to estimate the power spectral densities of the EEG signal and artifact signal can complicate the computational process. In addition to the techniques above, there are numerous other efficient strategies, including CCA, empirical mode decomposition (EMD), and sparse decomposition methods. Moreover, hybrid approaches combining these preprocessing algorithms with others, such as EMD-BSS, wavelet-BSS, and others, have been employed to maximize the algorithm's efficiency[143][144]. Additional information is available in previous publications[145][146][129].

2.5 Feature Extraction

The objective of feature extraction is to identify characteristics of the user's current activity and represent them as a feature vector. EEG data is too complicated and high-dimensional to manage a BCI without reduction; feature extraction effectively eliminates irrelevant data while discarding irrelevant data. BCI employs numerous distinct types of feature extraction. They can be divided into numerous categories, such as time-domain, frequency-domain, and spatial characteristics. Switching between the time and frequency domains can be accomplished using decomposition methods such as the Discrete Fourier transform (DFT) and the inverse discrete Fourier transform (IDFT), as well as algorithms referred to as fast Fourier transforms (FFTs) that enable the DFT to be calculated much more quickly. Analyzing a signal plotted in the time-domain reveals how the signal evolves over time and consequently enables the observation of time-dependent phenomena, such as the P300 wave, which arises 300ms after stimulus start and is only observable in the time-domain. Analyzing an EEG signal in the frequency domain yields no temporal information; rather, it reveals the proportion of a signal that resides in a certain frequency band relative to a number of given frequencies. This can be used to identify SSVEPs, if a time window of the right size is selected.

2.5.1 Amplitude Features

The amplitude of a signal can be used to train a classifier, such as with the P300 wave detector in the P300 speller.

2.5.2 Band power Features

As a feature, the average power of a signal inside a given frequency band is known as a band power feature. This is determined by bandpass filtering the signal and calculating the mean absolute value inside the band. By applying the log-transform to these features [42], which approximates the normal distribution, one can create a new type of feature known as log-band power features. Either feature type may be utilized to train a classifier for a BCI.

2.5.3 Power Spectral Density Features

Using the DFT to translate a signal into the frequency domain enables the production of useable BCI characteristics. By squaring the power spectrum and utilizing the values at the frequency of interest to train a classifier, it is possible to determine the power spectral density (PSD) characteristics. This approach is applicable to motor imaging, SSVEP[147][148][149], and numerous other BCI applications.

Canonical Correlation Analysis: CCA detects the association between two multidimensional variables and can be utilized for unsupervised SSVEP detection on EEG data[150][134][151][152][153]. Given two multidimensional variables X and Y with weighted linear combinations $x = X^T W_X$ and $y = Y^T W_Y$, CCA determines the weight vectors W_X and W_Y that maximize correlation between x and y. This is accomplished by addressing the subsequent optimization problem:

$$\begin{aligned} \max_{W_X, W_Y} \rho(x,y) &= \frac{E[xy]}{\sqrt{E[xx]E[yy]}} \\ &= \frac{E\left[W_X^T X Y^T W_Y\right]}{\sqrt{E\left[W_X^T X X^T W_X\right] E\left[W_Y^T Y Y^T W_Y\right]}}, \end{aligned} \tag{2.15}$$

where $E[x]$ is the expected value of x and ρ is the correlation value that is maximized with regard to the weight vectors W_X and W_Y to calculate the canonical correlation between X and Y.

$\mathbf{X} \in R^{C \times S}$ is the multidimensional EEG signal with C channels and S samples during SSVEP detection. $Y_f \in R^{2N_h \times S}$ is the collection of multidimensional reference signals based on the stimulus frequency f, with $2N_h$ sine waves and S samples, where Nh is the number of harmonics. Sine waves are arranged in a matrix [60]:

$$Y_f = \begin{bmatrix} \sin(2\pi ft) \\ \cos(2\pi ft) \\ \dots \\ \sin(2\pi N_h ft) \\ \cos(2\pi N_h ft) \end{bmatrix} \tag{2.16}$$

where t represents time in seconds. By running CCA on X and Y_f for every f, the stimulation frequency with the greatest canonical correlation value may be discovered; this frequency is then chosen as the estimated SSVEP frequency.

Common Spatial Pattern Features: Method previously described for training spatial filter W, which filters EEG data into spatially filtered signal Z. As the rows of Z are separated maximally based on the variance between classes one and two, the outermost m rows are chosen. Log-variance features can be retrieved from the signal $Z_p(p = 1, ..., m)$ using:

$$f_p = \log\left(\frac{\text{var}(Z_p)}{\sum_{i=1}^{2m} \text{var}(Z_i)}\right) \tag{2.17}$$

where f_p is the feature vector $(1X2m)$. The log transformation approximates the normal distribution, and the characteristics can be utilized to train a classifier and predict the class of fresh data.

2.6 Feature Extraction Methods

2.6.1 Principal Component Analysis

PCA is a widely used linear transformation that reduces dimension. PCA introduces a vector in a lower-dimensional space to reduce the time- and space-dependent signal complexity[154]. PCA is used to isolate artifacts from actual signals, however this transformation can also be utilized to extract features without losing information[**?**]. PCA generates a set of non-correlated linear vectors (i.e., principal components) by transforming correlated variables from the original signal[**?**]. Although principal components improve signal similarity and data classification performance[**?**], they are less interpretable than fundamental features. In addition, PCA is inadequate for analyzing complicated data sets[**?**]. Several variations of PCA, such as kernel PCA[**?**] and sparse PCA[**?**], have been proposed in EEG data processing to counteract these shortcomings.

2.6.2 Autoregressive Mode

The AR model is a method for feature extraction in frequency domain analysis that has been used to analyze non-stationary signals such as EEG data[154]. AR posits that the AR process can anticipate genuine EEG signals; this prediction can be made using the order and parameters of the approximation model. The order of the AR model is a number between 1 and 12, which shows the model's performance. Choosing a suitable value for the order of the AR can be difficult due to the fact that wrong order selection leads to incorrect spectrum estimate, which may raise the computing costs[**?**]. AR approaches such as bilinear AAR, adaptive AR parameters, and multivariate AAR have been utilized extensively in EEG data processing, allowing AR model parameters to adjust to nonstationary EEG signals[155]. These strategies contribute to successive parameter estimation and the reduction of prediction error. Using the Kalman filter to evaluate AR parameters in an adaptive AR model, for instance, can improve classification performance by up to 83%[156]. Other advantages of the AR model include its suitable frequency resolution[157] and its applicability in estimating the power spectra

of shorter EEG data segments[158]. However, AR is susceptible to improper parameter selection and ordering[157].

2.6.3 Fast Fourier Transform

FFT is a reliable technique for stationary signals. It implements spectral analysis by transforming signals from the time domain to the frequency domain[159]. In this procedure, features are extracted by calculating the PSD using mathematical tools. FFT, which employs non-parametric approaches such as Welch's method[159][160], can be used to estimate PSD for a related band. Although FFT is frequently used in data analysis and is effective for stationary signals, it is ineffective for nonlinear and nonstationary data, such as EEG signals, and its conclusions are unreliable. This deficiency has inspired the development of novel processes and methods for the analysis of nonstationary signals, such as the Fourier decomposition method[161], the variational mode decomposition (VMD) approach[162], and the Hilbert-Huang transform (HHT)[163].

2.6.4 Wavelet Transform

WT is a time-frequency transform that takes into account the characteristics of EEG signals in the frequency domain and is completely localized in the time domain[164]. This approach performs well in the spectrum analysis of irregular and nonstationary signals in windows of varying sizes[165]. WT has the advantage of providing precise frequency and timing information at low and high frequencies, respectively. In other words, a narrow window is often utilized to analyze high frequencies, and a broad window is used to evaluate low frequencies[166][167]. Thus, WT is appropriate for transient oscillation in signals, especially biosignal data, which consists of low-frequency components with long-time periods and high-frequency components with short-time periods[156]. However, WT is negatively impacted by Heisenberg uncertainty, which hinders its performance[156]. WT examines small wavelets within a particular frequency range for a short time. Beginning at 0, the oscillations of the wavelets grow and subsequently drop back to zero [168].

2.6.5 Common Spatial Pattern

CSP is a successful method for feature extraction in BCI applications, particularly motor imaging tasks[169], and can be utilized for spatial filtering by employing the entire data trail or by dividing trails into temporal segments. CSP is commonly applied to binary classification applications[170]. CSP aims to distinguish between classes by decreasing the variance of one class and maximizing the variance of the other. This procedure can be carried out by incorporating spatial filters for each class. Using this method, EEG data are converted into a variance matrix that represents class discrimination[171]. The primary advantage of utilizing CSP is its simplicity and speed of execution[171]. However, there are intrinsic limits to this strategy for identifying optimal features from raw EEG data[169]. Several studies have developed optimal spatial feature selection strategies to overcome this issue. Using Dempster-Shafer theory and factoring in feature distribution, Jin et al.[169] have devised a unique technique for selecting features based on an enhanced objective function and considering feature distribution. Moreover, CSP is extremely sensitive to artifacts in the original data set, and modifying the electrode placements affects classification performance[172][173]. According to reference[172], a number of characteristics, including the frequency band filter, the time segment, and the subset of CSP filters to be employed, must be addressed in order to develop an effective CSP algorithm. Consequently, the performance of the CSP algorithm is contingent upon the subject-specific frequency band. Common spatio-spectral pattern (CSSP), common sparse spectral-spatial pattern (CSSSP), spectrally weighted common spatial pattern, sub-band common spatial pattern, and discriminant filter bank common spatial pattern have been proposed to address the problem of identifying the optimal frequency band for CSP algorithms. CSSP employs a simple time delay embedding with the CSP algorithm, which enhances the algorithm's performance by optimizing the frequency band at each electrode site [94]. However, the non-stationary EEG data diminish the CSSP's efficacy. Cho et al.[174] have added a noise reduction term in the Rayleigh coefficient of CSSP and devised an invariant CSSP algorithm that is both consistent and noise-resistant to address this dif-

ficulty. The invariant CSSP has the disadvantage that the optimal noise reduction value must be found. CSSSP[175] has been proposed to improve CSSP's performance. In contrast to the CSSP, which identifies distinct spectral patterns for each channel, this algorithm[176] seeks a spectral pattern shared by all channels. Sub-band common spatial pattern[177] is an additional extension of CSP used to filter EEG signals at several sub-bands in order to collect CSP characteristics from each sub-band, irrespective of the correlations between features from various sub-bands[176]. In order to circumvent this constraint, a discriminant filter bank common spatial pattern[178] employing the Fisher ratio of single channel band power values was also developed. Additional information regarding the CSP's scope of application can be found in[176].

2.7 Feature Selection

Feature vectors generated by feature extraction frequently require additional reduction, which can be accomplished by feature selection methods. This lessens the consequences of an issue known as the "curse of dimensionality"[179], in which the necessary amount of training data grows exponentially with the size of the feature vector. Other advantages include shorter training times, reduced storage needs, enhanced prediction performance, and the facilitation of data visualization[180]. Feature selection techniques can be categorized as filters, which choose subsets of variables as a preprocessing step, and wrappers, which evaluate various subsets of variables using the selected classifier[181]. Using techniques such as assessing the correlation between the variable and the target, filtering methods aim to select the best individual features. Wrappers evaluate subsets of features, which means they consider relationships between features. Effective application of both methods results in a model that can correctly classify a large subset of the data without 'overfitting'[181], where 'overfitting' refers to the phenomenon that occurs when a trained model becomes too fixated on a small number of data points to accurately classify new data. Forwards and backwards stepwise selection are typical wrapper method implementations. In forward selection, the model begins with a single feature and successively adds one variable at a time, as

long as the accuracy improves [67]. Backwards stepwise selection operates identically but in reverse, beginning with a complete set of features and removing them one by one.

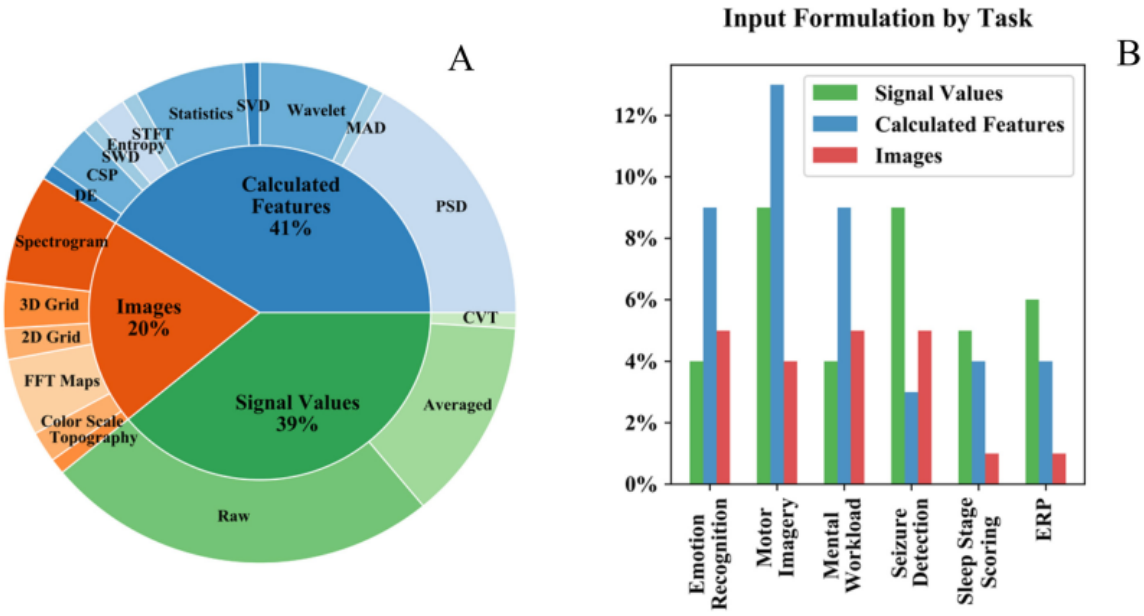


Figure 2.7: (A) Input formulations for each and every study. The inner circle indicates the general input formulation, whilst the outside circle indicates more detailed options. (B) A comparison of general input formulation across several tasks. Inputs for the majority of tasks were calculated features, but research on seizure detection contained a substantially higher amount of signal values. Key— CVT stands for complex value transformation, CSP stands for common spatial pattern, DE stands for dynamic energy, FFT stands for fast Fourier transform, MAD stands for mean absolute difference, PSD stands for power spectral density, STFT stands for short time Fourier transformation, SVD stands for singular value decomposition, and SWD stands for swarm decomposition.

2.8 Classification

Classification infers the present state of a user based on the feature vectors generated during feature extraction. Classification methods can be categorized into two groups: supervised learning, in which an algorithm is shown labeled samples of each class and then learns the classes to identify them later, and unsupervised learning, in which an algorithm is given unlabeled data and determines which categories best represent the data. The classification of BCI-controlled robots leads to a predetermined action (e.g. movement). There are numerous classification techniques available, such as artificial

neural networks, support vector machines, linear discriminant classifiers, and many others. In the BCI literature, linear discriminant analysis (LDA[182]) and support vector machine are two of the most used classification algorithms (SVM [183]).

2.8.1 Linear Discriminant Analysis

A common supervised learning approach, linear discriminant analysis separates classes using hyperplanes that maximize class separability. LDA states that the separation between classes is equal to the ratio of variance between classes to variance within classes[182]. LDA operates by identifying the weight vector w that maximizes:

$$J(w) = \frac{w^T S_B w}{w^T S_W w} \tag{2.18}$$

where S_B represents the scatter matrix across classes and S_W represents the scatter matrix within classes. LDA presupposes a normal distribution and equal covariance between classes.

2.8.2 Support Vector Machine

A support vector machine (SVM[183]) is another supervised classification method that separates classes using a hyperplane; however, the emphasis is on maximising the distance between the outer margins and the closest training data points on either side of the hyperplane (Fig. 2.7), which are referred to as the support vectors. SVM can be extended to non-linear classification via the 'kernel trick' [184], despite its initial use as a linear classification approach.

2.9 Conclusion

This chapter examined in depth the fundamental processes that allow a BCI to function, as well as the diverse ways for implementing a BCI. There are numerous varieties of BCI, and methodologies can differ in terms of signal detection and synthesis, feature extraction and selection, classification, and translation; each has its own advantages

and disadvantages. Although various neuroimaging techniques are BCI-capable, EEG is the most practical because to its low cost and portability. Due to the strength of the SSVEP response, SSVEP-BCIs are chosen for a fast-response BCI, and CCA-based detection methods are appropriate for this purpose due to their accuracy and lack of training data requirements. In contrast, motor imagery BCIs require no external input and can be discovered using CSP-based techniques.

Chapter 3

Machine and Deep Learning for EEG

3.1 Introduction

This chapter introduces the use of machine learning in the preprocessing of EEG signals and their classification. Furthermore, we discuss several machine learning and deep learning algorithms to classify EEG signals. In this chapter, we provide a technical background on ML/DL and its hyperparamaters to be implemented in EEG classification and decoding.

3.2 Machine Learning Overview

The application of a collection of mathematical models and algorithms to steadily enhance the performance of a single task is machine learning. It uses training data sets as input to generate estimates without being explicitly programmed to do so. In this area, the tasks are extremely diverse and can be divided into two primary categories: supervised and unsupervised learning. Unsupervised learning occurs when an algorithm constructs a recognition pattern from a data set having only inputs and no predetermined outputs. Semi-supervised learning is a subset of supervised learning. They are identical in that they both learn from data sets with known inputs and outputs, with the exception that semi-supervised is missing portions of the data set. Supervised learning is typically employed in classification and regression applications, but unsupervised learning lends itself to feature learning and the opposite, dimension reduction. This study will examine some of the most prevalent machine learning algorithms and classify them according to the type of learning, with applications to EEG[185].

With the help of machine learning techniques, EEG waves can be used as markers

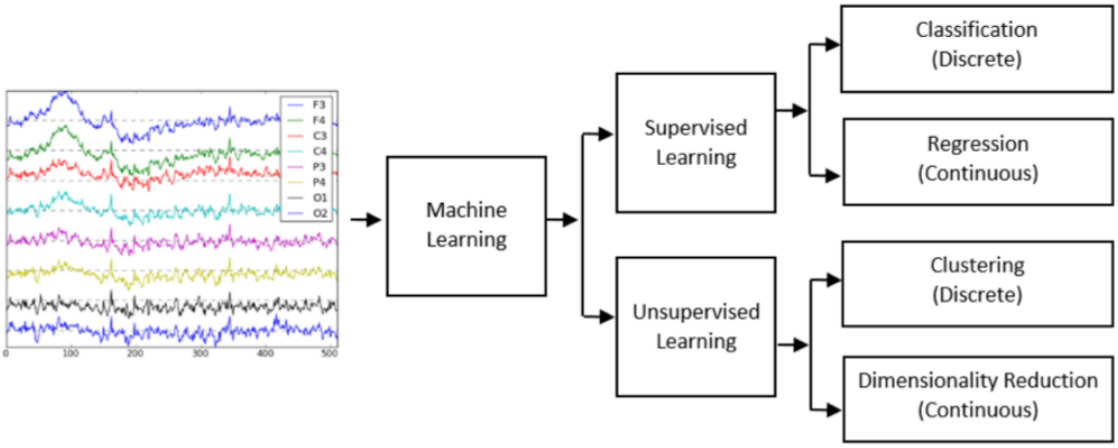


Figure 3.1: Based on supervised and unsupervised learning found in the literature, machine learning applications for EEG have been created. The subcategories of supervised learning are classification and regression, which generate discrete and continuous outcomes, respectively. Unsupervised learning is divided into clustering and dimensionality reduction, which generate discrete and continuous outcomes, respectively.

of difficult-to-detect medical disorders. The applications of machine learning on EEG signals based on supervised and unsupervised learning are depicted in the image below. Supervised learning creates a prediction model based on both input and desired output data, which is categorized into classification and regression, which provide discrete and continuous results, respectively. Unsupervised learning creates a predictive model by categorizing input data according to clustering and dimensionality reduction in order to generate discrete and continuous variables.

3.2.1 Regression

Regression modeling is a popular statistical technique because it provides a straightforward method for establishing a functional link between variables. Univariate and multivariate for quantitative response variables; simple and multiple for predictor variables; linear for linearly transformable data; nonlinear for nonlinearly transformable data; analysis of variance for qualitative variable predictors; analysis of covariance for the combination of qualitative and quantitative variable predictors; and logistic for qualitative response variables [186]. Legendre and Gauss were the first to use regression using the Least Squares Method. In Linear Regression, this method pro-

duces approximations by summing the squares of each equation's residual to best fit the data[185]. Linear Regression is one of the most widely used regression methods. In this approach, the parameters are defined as a linear combination, despite the fact that each independent variable is not always linear. Multiple linear regression is analogous to simple linear regression, with the addition of multiple independent variables. Non-linear regression must be applied when the parameters are not linear. This minimizes the function using a sum of squares technique, but with an iterative procedure[185].

3.2.2 Support Vector Machine

SVM is a subclass of supervised learning that is utilized to analyze data for classification and regression. The objective is to map points in space so that examples of the target categories are separated by the greatest margin possible. This enables SVM to have a smaller generalization error as a classifier in general[187]. In an N-dimensional space, the goal is to locate a hyperplane or set of hyperplanes. Support vectors are the data points closest to a certain hyperplane. By adjusting the position and orientation of the hyperplane, they maximize the classifier's margin. Due to the location of the data, it is also possible that the points in this space cannot be separated linearly. SVM is possible to apply created kernel functions, often known as "kernel trick," to the data set in order to resolve this issue. This approach requires transforming the existing algorithm from a lower-dimensional to a higher-dimensional data set[185]. In this higher-dimensional space, it is possible to develop a linear classifier despite the fact that the amount of data remains unchanged. Multiple K kernels are assigned to each point in order to determine the optimal hyperplane for the transformed feature space. With sufficient K functions, it is possible to achieve accurate separation. The only significant issue is overfitting[188].

3.2.3 KNN K-Nearset Neighbours

KNN is one of the algorithms for supervised machine learning. In supervised learning, the relationship between input and output is already established in the training data set,

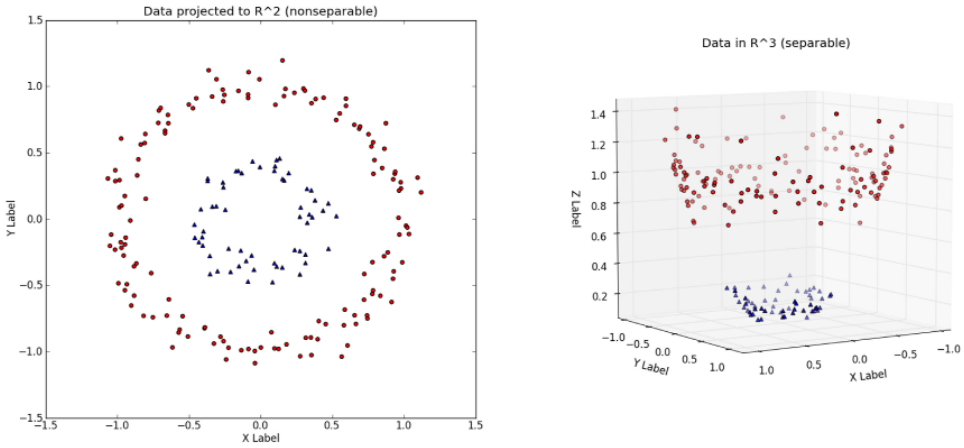


Figure 3.2: Higher dimension kernel separation. The kernel trick requires transforming the existing method from a lower-dimensional to a higher-dimensional data set.

i.e. the output for a given input is already known. The categories of supervised learning are regression and classification. KNN is applicable for classification and regression. The inputs for classification and regression are same, but their corresponding outputs differ[185]. Example input-output pairs are used to anticipate the output of a data set that has not been trained. KNN classifies the input based on its K neighbors' classifications. To discover the nearest neighbors, Euclidean distance or Mahalanobis distance is calculated between all known data points and the input. After calculating the distance, the K nearest neighbors are selected. The input is then categorized based on similarities between the input and its K-neighbors. The size of the data collection is used to determine the value of K. If the result of taking the square root of the size of the data collection is an even integer, 1 is added or subtracted. The outcome is then determined to be K for the given data set. K is chosen to be an odd number to prevent bias in input prediction[188].

3.2.4 Artificial Neural Networks ANN

Neural networks, often known as artificial neural networks in the computing field, is a mathematical model that closely resembles the neural network structure seen in the human brain. Several ideas and examples demonstrating the interaction between different layers of neural networks to transform the provided input into the desired output have

been proposed by researchers in an effort to explain how the model functions[185]. Imagine you are in a bar, perusing the menu for a tasty drink. You order IPA as soon as you see it on the menu since it is your favorite. So, what occurred in your brain was that you offered several inputs for beer choice to your brain's neural network, with IPA having a preferential weight as your favorite beer; the brain then made a decision and provided you with an output. This is a fundamental illustration of how neural networks function. The model's design illustrates the decision-making process, which consists of considerably deeper layers of interaction between the input and output layers[185].

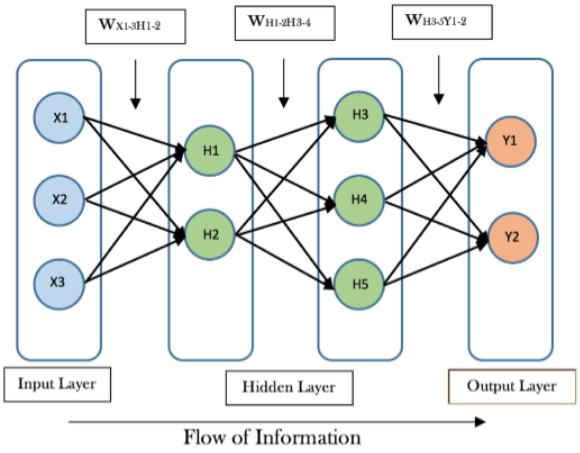


Figure 3.3: FeedForward Neural Network

As each application is unique and requires a specific technique – long-term or short-term EEG segment analysis, real-time or time-delayed process, type of EEG channel analysis (single or multiple) – these approaches may be readily targeted and synthesized with the help of ANN. Once the EEG signals have been translated to waveforms in user-friendly GUIs, they are classified using ANN, with a specific type of network selected for a given use case – Feedforward backpropagation, Radial basis function, or Recurrent Neural networks. It is essential to understand how various forms of ANN operate and the architecture that enables their operation.

1. **FeedForward Neural Networks:** This is a sort of network in which data flows in a single direction, beginning at the input nodes, passing via the hidden nodes, and ending at the output nodes. This network prevents the formation of loops and cycles, allowing information to flow in just one direction.

2. **Radial basis function:** RBF is a sort of artificial neural network (ANN) that uses radial basis functions in the fields of artificial neural networks and mathematical modeling (An arbitrary real-valued function, the value of which is determined by functions location from the origin). Thus, the output of the network is determined by a linear combination of RBF of the inputs and provided parameters for the neurons. As shown in Figure 5, the structure generates the final output by summing the centers/widths of the points with their corresponding weights.

3. **Recurrent Neural Networks:** As the name suggests, RNN is a sort of Artificial Neural Network having connections between distinct nodes and a defined output flow direction to a certain node. In this case, the data flow can form loops and cycles to return data to the intended node. This method is demonstrated

3.2.5 Naive Bayes

The popular Naive Bayes classifier employs Bayes' theorem to segregate data based on simple training features. Within a limited set, the model assigns labels as feature vectors. With sufficient preprocessing, it can compete with more advanced algorithms, such as SVM, mentioned above. The one shortcoming of the naive Bayes[185] technique is that it treats all feature vectors as independent from one another, regardless of any actual correlation. The primary advantage is that it just requires a limited number of training data sets to begin accurately estimating the classification parameters. The Bayes technique can be implemented with many models. The most prevalent is the prob- probabilistic model. In this paradigm, features are represented by vectors, and probabilities are assigned to each instance or result. Event models can be divided into two major categories: Gaussian and Multinomial Naive Bayes. A fair assumption for a data collection with continuous values is that it follows a Gaussian distribution. The Bayes technique assigns probability based on the curve using this method. A multinomial event model derived from multinomials, typically shown as a histogram. A potential cause for concern is the absence of a feature from the data set. This results in the multiple of all estimations being equal to zero. It can be rectified with a pseu-

docount to smooth out data set outliers[189]. Represents the frequencies of particular events generated by multinomials, typically as a histogram. A potential cause for concern is the absence of a feature from the data set. This results in the multiple of all estimations being equal to zero. It can be rectified with a pseudocount to smooth out data set outliers[189].

3.2.6 Decision Tree and Random Forest

To classify data, decision trees ask questions about the characteristics of an object. Each question can be represented as a node, and each answer can be represented as a child node. This results in the formation of a hierarchy, or a tree. The simplest tree would be one in which each inquiry yields a yes or no response. Therefore, there is a yes or no child node query for each parent node. The data is sorted through the tree by beginning at the root, also known as the node at the very top, and working its way down to the leaf, which is the node with no children. The path followed depends on the characteristics of the data. Once data reaches a leaf, it can be categorized according to the class associated with that leaf[190].The advantages of decision trees are that they are straightforward and may be easily coupled with various strategies for decision making. The downsides of decision trees include that they are relatively unstable as well as imprecise, especially with varied level sizes which generate biases towards higher levels. In the study of machine learning and various classification and distribution methods, we encounter the Random Forest methodology, which may be utilized for both data categorization and regression. As its name implies, Random Forest operates by generating a large number of decision trees and is trained by performing a bagging operation to merge many decision trees or models to produce more stable and accurate data predictions. Random Forest adds randomization to the data being organized; i.e., rather than identifying the most significant feature from the given set, it identifies the best feature from a random subset of features. This results in a more diverse and superior model of outcomes. In Random forest, the solutions from all trees are summed, and classification is determined by majority vote, with the most appro-

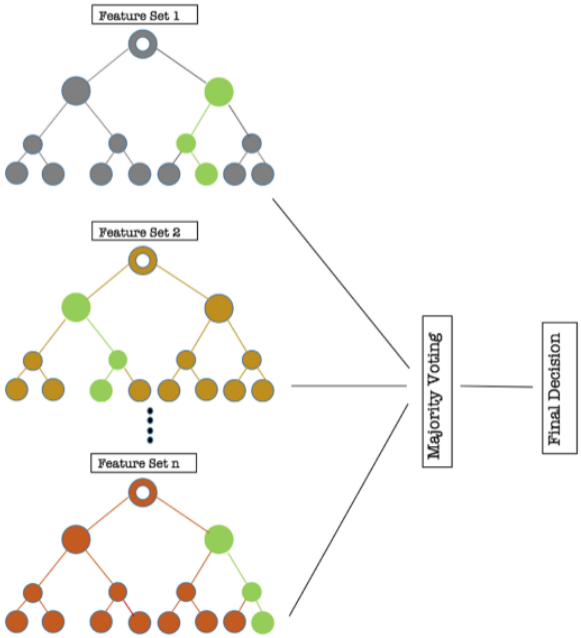


Figure 3.4: Random Forest as a type of ensemble learning primarily used for classification and regression

priate classification selected. Nonetheless, if the trees are proven to be unstable, such that little changes in the data set might alter the entire decision tree, we may arrive at an incorrect categorization[189].

3.2.7 Ensemble Learning

Ensemble learning is a form of supervised learning. As its name suggests, ensemble learning combines multiple algorithms to create a model with improved prediction performance. The general objective is to increase overall performance by merging judgments from many multiple models. Based on the concept of diversity, while obtaining results for the same problem, more diverse models are considered than single models. This provides a collection of theories that can be merged to improve performance. All of the individual models are referred to as basic learners when merged into an ensemble. The ensemble is generally superior to the learners from which it was formed. Ensemble learning can be implemented in various domains, including medicine, fraud detection, banking, malware and intrusion detection, face and emotion identification, etc[185].

3.2.8 Fuzzy Logic

Fuzzy Logic is utilized by the vast majority of household appliances and machines, like the air conditioner, washing machine, etc. This logic is given to a control system typically referred to as the Fuzzy system control, where each component is designed to operate and modify another physical operating system in order to achieve the desired functionality[185]. To comprehend how a fuzzy system operates, it is crucial to examine the system's requirements and its intended use[191]. A knowledge-based functional element with the capacity to apply human cognitive processes, such as reasoning and thinking, must have a stable component that can deliver output depending on the degree of truth for a particular set of input variables in order to constitute a system. For a fuzzy system to work, several components must be reliable:

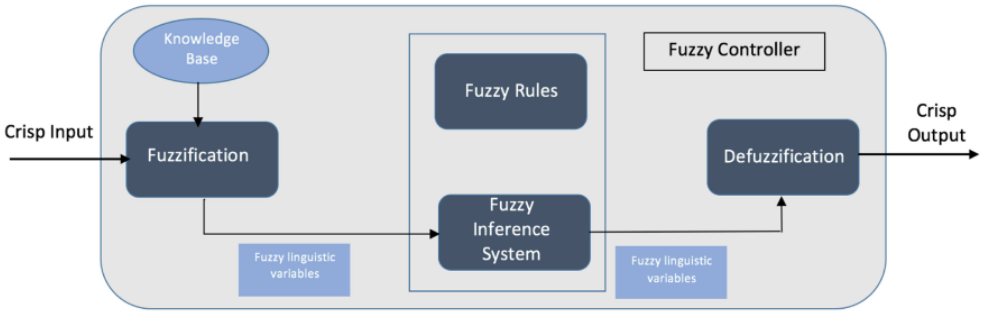


Figure 3.5: Example of a Fuzzy System

1. **Fuzzy sets:** A fuzzy set corresponds to a member function defined in a fuzzy space. A member function gives any element in well-defined fuzzy sets a membership degree. The member function assigns these elements a value between 0 and 1, where 0 means the element is not in the fuzzy set and 1 means it is.

2. **Fuzzy Rules:** A collection of fuzzy rules determines the output of fuzzy logic's IF-THEN rules. IF-THEN rules construct fuzzy logic conditional statements. IF-THEN assumes X and Y are intended terms and evaluated by fuzzy sets with range U and V. This separates antecedent and consequent. If the antecedent states X and U, the consequent should state Y and V. X is U if Y is V. These principles are based on natural language, fuzzy sets, and logic.

3. **Fuzzy Logic Inference or Fuzzy Inference System (FIS):** Once the fuzzy rules and membership functions are developed, the FIS is implemented for process simulation and control. FIS has 3 stages: First, numerical input variables are mapped for fuzzy set compatibility. This is Fuzzification. This technique translates input and output into fuzzy-readable language. Second, the system processes rules based on each input variable's strength. Defuzzification converts fuzzy values to numbers in the third stage. This method transfers fuzzy output to crisp, making it apparent.

4. **Fuzzy Score:** The FIS system outputs a fuzzy score for all known input scores. FIS generates the fuzzy score by considering all fuzzy restrictions and membership functions. Score depends on applied rules and input factors. The FIS scores every input variable using fuzzy rules.

As pattern identification of EEG data is the primary application of Machine Learning, Fuzzy Logic can be utilized to determine the proper recognition rate of EEG classifications at various stages. However, a combination of Fuzzy logic and Neural networks, typically referred to as the Neuro-Fuzzy system, is utilized. This system applies fuzzy parameters (such as fuzzy sets and fuzzy rules) and combines them with neural network approximation techniques for extended analysis. For medical condition diagnosis, density and regression estimates, pattern recognition, and data analytics, the Neuro-Fuzzy system[192] has been found to be extraordinarily effective[185].

3.2.9 Linear Discriminant Analysis

For a particular data set including a large number of random variables, it is important to execute dimensionality reduction to decrease the number of parameters to certain principal variables in order to lower the dataset's dimensional space. As there are numerous ways to classify data, the dimensionality reduction methodology is accomplished using two methods: the Principle component analysis and the linear discriminant analysis. PCA and LDA have functions and applications that are comparable. However, the LDA method can handle circumstances in which the within-class frequencies do not

need to be equal, and it gives a high ratio and large separation between between-class variance and within-class variance. The primary distinction between PCA and LDA is that PCA is more relevant for feature classification, whilst LDA is applicable for data classification. Linear discriminant analysis is the most often used method for dimensionality reduction (LDA). This technique's primary criteria are to provide effective separation between distinct classes and to prevent curve overfitting. By projecting the supplied feature space with n-dimensional samples onto a more precise and smaller featuresubspace, this greatly reduces processing costs and improves categorization. In a typical PCA analysis, the location, shape, and structure of the data set undergo radical transformations. In contrast, the LDA approach preserves the location and shape of the data set as it is translated into a smaller space. This is achieved by establishing a collection of vectors to distinguish and separate on the modified space[185].

3.2.10 K-Means

K-means is a method of unsupervised learning used to solve the clustering problem. Using an algorithm, it finds a partition that minimizes the difference between a cluster's empirical mean and points within. K-means uses these K clusters to attempt to minimize the sum of squared errors[193]. There are two typical initialization methods: Forgery and Random Partition. The Forgy approach selects K observations at random from the data collection. Then, these observations serve as the initial mean. For the Random Partitioning approach, random clusters are initially allocated to each observation. This is then changed as the initial mean is computed to be in the cluster's center. Given that K is relatively small, one of the benefits of K-means is its easy implementation of high computing speed. The substantial significance of initial circumstances on final outputs, sensitivity to scaling, and a link between data order and end findings are all downsides of K-means[185].

Features identification	Extraction	Time-domain	Autoregressive (AR) Adaptive autoregressive Root-mean-square (RMS) Integrated EEG (IEEG)
		Frequency-domain	Fast Fourier transform (FFT) Welch's method Local characteristic-scale decomposition (LCD)
		Time-frequency domain	Short-time Fourier transform (STFT) Wavelet transform (WT) Discrete wavelet transform (DWT)
		Spatial domain	Common spatial pattern (CSP) Common spatio-spectral pattern (CSSP) Common sparse spatio-spectral patterns (CSSSP) Sub-band common spatial pattern (SBCSP)
	Selection	Statistical transformation	Principal component analysis (PCA) Independent component analysis (ICA)
		Filter bank	Filter bank CSP (FBCSP) Discriminant filter bank CSP (DFBCSP) Sparse filter bank CSP (SFBCSP)
		Evolutionary algorithms (EAs)	Particle swarm optimization (PSO) Differential evolution (DE) Artificial bee colony (ABC) Ant colony optimization (ACO) Genetic algorithms (GAs)
	Classification	Linear	Support-vector machines (SVMs) Linear discriminant analysis (LDA) Naive Bayes Logistic regression (LR)
		Nonlinear	Bayes quadratic Hidden Markov model (HMM)
		Nearest neighbor	k-nearest neighbor analysis (k-NN)
		Artificial neural network (ANN)	Multilayer perceptron (MLP) Radial basis function (RBF) Deep neural network (DNN)

Figure 3.6: An overview of signal processing approaches for the extraction, selection, and categorization of characteristics.

3.3 Machine Learning for EEG Classification

Artificial intelligence encompasses machine learning, and deep learning is a rapidly expanding field with applications in classification [99]. Classification's purpose is to predict the class label of fresh data pieces for a variety of activities [27]. Classification algorithms can be categorized into two groups: traditional classification algorithms and DL algorithms [100]. Using input data and statistical analysis to categorize output values, conventional classification algorithms construct classification models with great precision. The majority of conventional classification techniques include hand-crafted

input features for model training. This process, known as feature creation, is limited in its ability to process input from high-dimensional data sets [100]. DL techniques rely on representation learning [101] and can address the limits of standard classification algorithms by automatically learning features at several abstraction levels [100]. Among standard classification algorithms such as supervised learning and unsupervised learning, supervised algorithms are the most well-known technique for analyzing EEG data [102]. Artificial neural networks are one of the most widespread supervised algorithms (ANNs). ANNs are computational models [103] that employ multi-layered neural networks with weighted connections between units, often followed by a static non-linearity function (e.g., ReLu). During the learning phase, the network can alter its weights to improve its classification performance on test data [104]. Similar and well-known supervised techniques include naïve Bayes (NB), support vector machine (SVM), k-nearest neighbor (KNN), logistic regression (LR), random forest (RF), and linear discriminant analysis (LDA). Each supervised model uses a learning algorithm to produce an increasingly accurate model [105]. NB is a probabilistic classifier that employs Bayes' theorem to categorize data based on particular characteristics [106]. It is a straightforward and effective classifier that requires just short training data sets to estimate the classification parameters. This advantage makes NB a robust classifier for the study of EEG data in a variety of tasks, including ER [107], seizure detection (SD) [108], and MI [109]. NB is based on the premise that all qualities are independent of one another and that all feature vectors have the same effect on the outcome [106]. SVM has been shown to be a good supervised model based on a statistical learning instrument with a high degree of generalization. SVM is based on the notion of separating two data sets. This division may be linear or nonlinear. In the case of linear separation, SVM distinguishes classes using a discriminant hyperplane. In the event of non-linear separation, however, SVM employs the kernel function to determine decision boundaries. In comparison to other supervised algorithms, such as ANNs and KNN, SVM has a low computational complexity [110, 111]. Although increasing the k-value lessens the computational cost of KNN, it also affects its classification ef-

ficiency [110,112]. In addition, with the development of DL algorithms, SVM has continued to be frequently employed in EEG signal categorization due to its mathematically sound computation. Nevertheless, the effectiveness of SVM is dependent on the kernel function and penalty coefficient parameters; improving the parameters included into SVM classifiers is crucial [113]. Huange et al. [114] have used a genetic algorithm to optimize SVM parameters, and Wang et al. [115] have proposed particle swarm optimization. According to our investigation, SVM has been widely employed in the categorization of EEG signals due to its simplicity and adaptability in handling classification problems such as the diagnosis of brain illnesses (e.g., SD and Alzheimer's disease) [116–118]. RF is a supervised tree-based technique that builds an ensemble of decision trees. During the training phase, each decision tree is produced. RF draws predictions from each tree and determines the final conclusion via a voting technique or by averaging the findings [119] in order to determine the class that is most frequently utilized. This and related ensemble methods are based on the premise that a set of weak classifiers can generate a strong classifier to create a successful learning algorithm. However, the overfitting and instability of trees might impact the performance of RF models, especially with trees of various sizes [106]. Unlike the LR model, which is a probabilistic classification model for both binary and multi-class classification tasks [120], the RF model operates on both discrete and continuous data, hence giving models for classification and regression problems. In addition, the parallelization structure of RF enables it to outperform other supervised algorithms on large EEG data sets when handling classification challenges [109]. LDA is a linear transformation approach that is used to determine the linear combinations of variables that most effectively distinguish classes [121, 122]. LDA is predicated on the assumption that the data density is normally distributed, with equal covariance between classes. Obtaining the separating hyperplane is accomplished by maximizing the distance between the two classes while minimizing the distance between points within each class [123]. This method is simple to implement and requires very little processing. As a result, LDA has been effectively used to handle classification issues in BCI systems

such as MI-based BCI [124], P300 speller [125], and multiclass BCI [126]. However, the most significant restriction of this model is its linearity, which hinders competitive findings on nonlinear EEG data [127,128].

3.4 Deep Learning

In-depth analysis is required to extract all the useful information from huge datasets. Traditional classification techniques perform poorly with big, dynamic datasets because they cannot account for all variety states within the data [18]. DL is a subfield of machine learning that aims to emulate the activity of the human brain by constructing a sophisticated, interconnected neural architecture and, as a result, retrieving a generic model capable of handling several input sources. DL strives to overcome the limitations of conventional neural networks by including all information within a training dataset. DL is an end-to-end methodology [78], which means that raw data can be supplied directly into the DNN for learning the parameters and hyperparameters. Alternatively stated, DL allows DNNs to be fed raw data with minimal or no preprocessing; additionally, DL performs feature extraction, selection, and classification as a single pipeline. In addition, the DL method is regarded as universal, robust, general, and scalable [35]. However, training DNN requires a large number of parameters and hyperparameters, which extends the training time relative to other methods and consumes significantly more hardware resources [71,79,80]. Nevertheless, it is possible to address these issues using other processing technologies, such as GPUs.

3.4.1 Architecture design choices

CNN's (43 percent) architecture design framework consists of alternating layers of convolution and pooling layers (typically maximum pooling layers). The number of convolutional layers and the type of final classifier were the most important design elements of CNNs. 18 percent of viewers choose DBN after CNN as the second most popular option. DBNs consist of a number of stacked limited Boltzmann machines followed by an end classifier, which consists of a number of typically fully-connected

layers. 12 percent of the total number of studies comprised hybrid architectures, which were separated, as indicated in the figure below, into two categories: hybrid CNNs and hybrid MLPs. In addition to convolutional and pooling layers, hybrid CNNs incorporate an extra architecture type, such as recurrent layers or limited Boltzmann machines. Hybrid MLPs consist of multiple dense layers and another sort of deep learning algorithm. RNNs (10 percent of the total number of studies) are formed of recurrent layers (each layer containing a study-specific number of recurrent units) followed by fully-connected layers. MLPNNs, whose sole evaluated characteristic was the number of hidden layers, accounted for 9 percent of the total number of studies. The third group of research (8 percent) utilized an SAE, which consists of the entire number of fully connected levels followed by a single fully linked layer in every case.

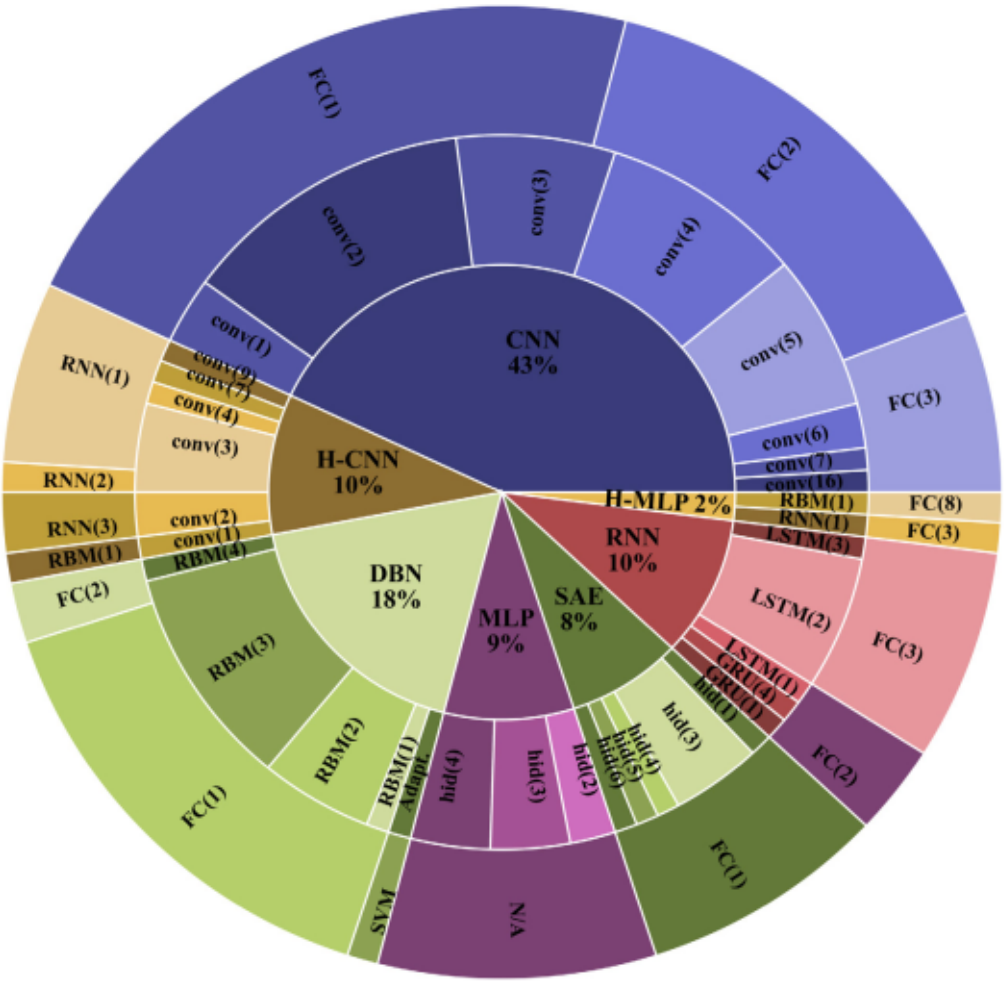


Figure 3.7: Deep learning architectures across all studies.

3.4.2 Activation functions

All research collected activation functions for appropriate deep learning architectures. Seventy percent of research adopting convolutional layers for deep learning architectures employed rectified linear unit (ReLU) as the activation function for the layer's convolutional layer. No other activation function surpassed 10 percent of all studies that reported activation functions. Eight percent of activation functions are exponential linear unit (ELU), eight percent are leaky rectified linear unit (leaky ReLU), and eight percent are hyperbolic tangent (tanh) (5 percent). In addition, there were individual experiments employing the activation function types parametric ReLU (PReLU), scaled exponential linear unit (SELU), and split tanh. In the discussion section, further examination of convolutional activation functions is elaborated. Non-classifier fully-connected layers and classifier fully-connected layers can be categorized according on their activation functions. The vast majority of classifier fully-connected layers used the softmax activation function, while non-classifier fully-connected layers employed the sigmoid activation function. Only three SAE research examined activation functions, and there was no unanimity among them; [194][15] used sigmoid activation functions for non-classifier AE layers, whereas[195] employed ReLU. To further comprehend the most effective activation function for SAE architectures, more research is required.

3.4.3 Task specific deep learning trends

On tasks involving emotion identification, motor imagery, and sleep stage scoring, there was no unanimity about the selection of deep learning algorithms. Studies on seizure detection utilized either CNNs or RNNs, with the highest proportion of studies employing RNNs relative to other tasks. Only one of these investigations used an SAE or MLPNN, and none of the seizure detection experiments used DBNs. Sleep stage scoring tasks had the highest proportion of research employing hybrid formulations, which were equally represented in comparison to studies employing CNNs[15]. ERP studies clearly favored CNN's coverage (the highest percentage of CNN studies

compared to all other tasks). Decisions on task-specific deep learning strategies are depicted in the image below[14].

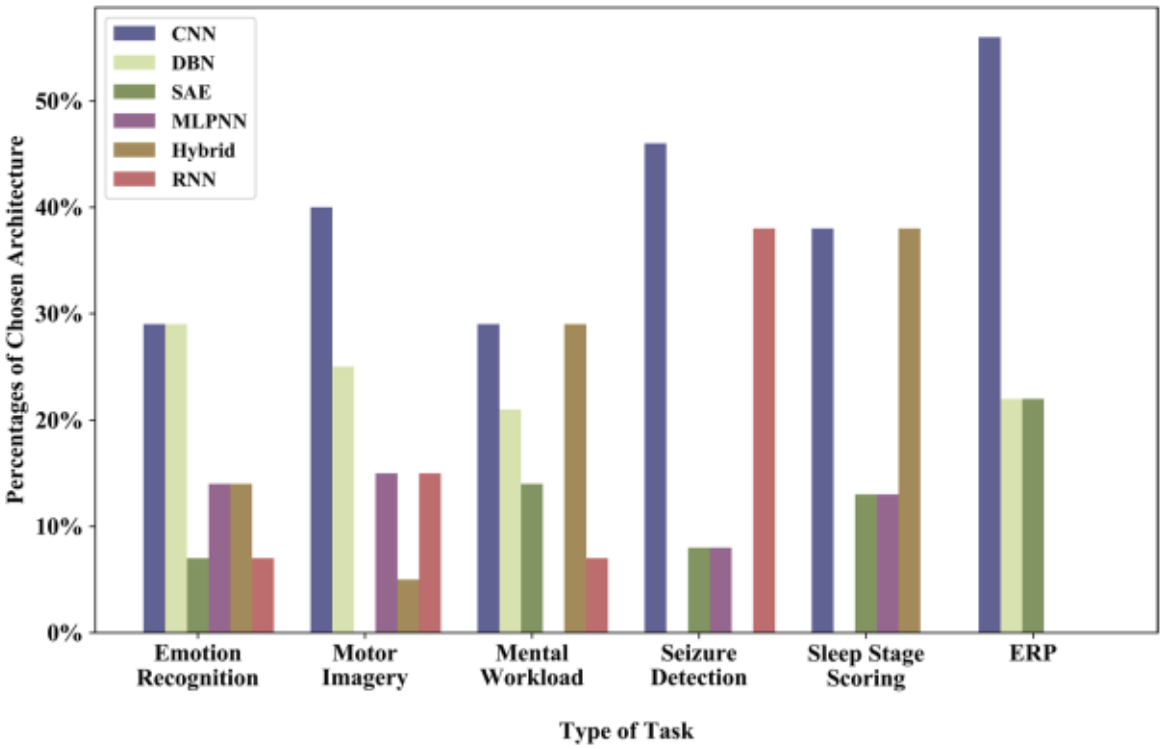


Figure 3.8: The proportions of deep learning architectures by task type[14].

3.4.4 Input formulation by deep learning architecture

Depending on the type of deep learning architecture, the specific input formulation methodologies differed substantially. It is hardly unexpected that none of the DBN, MLPNN, or SAE experiments employed images as inputs, since image processing is regarded to be the realm of CNNs. The average accuracy of CNN experiments that used photos as inputs was comparable to CNN studies that used calculated features as inputs, with both input formulation methodologies obtaining an average accuracy of 84%. In comparison, the average accuracy of CNN experiments employing signal values as inputs was 87%. This contradicts the common belief that the more time invested in the pre-processing stages, the more precise the categorization will be. This leads to the unexpected conclusion that future research, rather than being hampered by the desire to spend less time on preprocessing, may instead improve outcomes by

passing signal values directly to the deep learning framework[14].

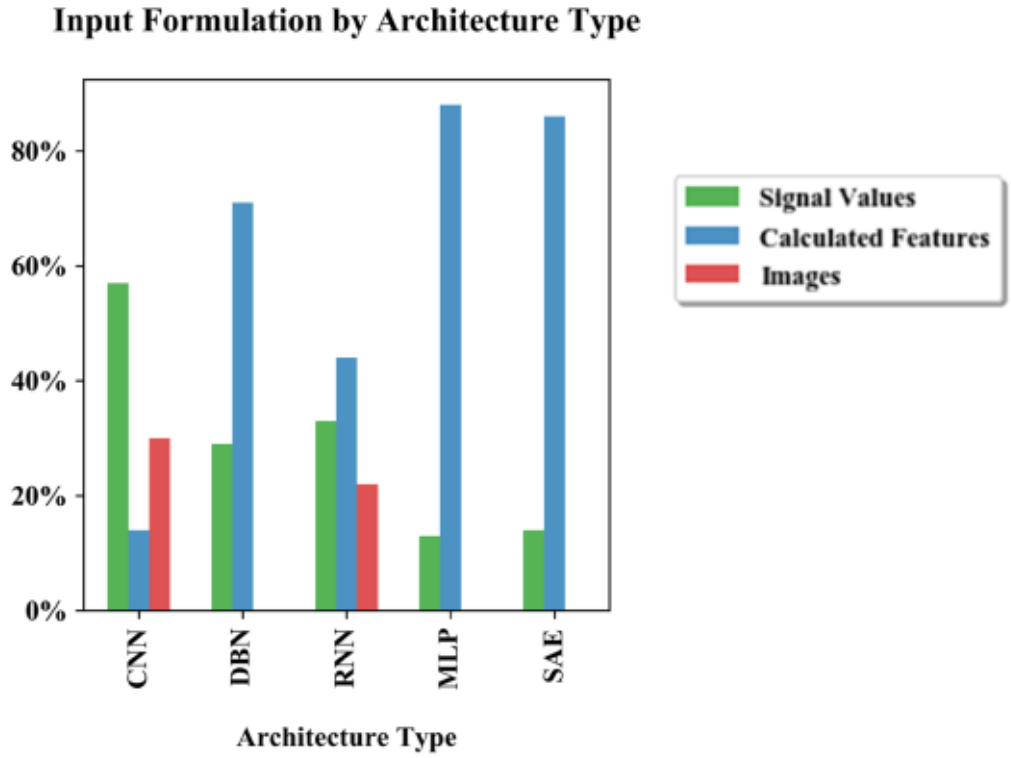


Figure 3.9: The proportions of input formulations by deep learning architecture types.

The DBN studies revealed a similar tendency to the CNN studies. Using signal values or calculated features in DBN investigations was a toss-up, with calculated features being the more popular choice. Studies utilizing computed features achieved an average accuracy of 85%, whilst studies utilizing signal value achieved an average accuracy of 86%. No studies compared the classification accuracies of signal values and computed features, indicating that additional research is required. In contrast, RNNs did not exhibit the same pattern as CNNs and DBNs. All three types of picture formulations were represented by a similar amount of examples in RNN research. The average accuracy of RNN studies that used signal values as inputs was 85 percent, which is lower than the accuracy of studies that employed computed features (89 percent) and photos (100 percent). However, much fewer studies have utilized RNNs, therefore additional study is required to determine the best effective input formulation technique for RNNs[14]. Feature selection appeared to be the only viable option for MLPNN and SAE research, as just one study for each architecture opted to use sig-

nal values instead. The only MLPNN study to use signal values [55] attained a 75 percent accuracy rate[15]. The solitary SAE investigation employing signal values as inputs [53] had just one channel for processing, but was able to obtain an accuracy of 96%[14].

3.5 Deep Learning Models

In this section, we provide a formal introduction to the deep learning models, encompassing concepts, structures, and methodologies typically employed in brain signal research. Deep learning is a category of machine learning algorithms that employs multiple layers of information processing stages in hierarchical architectures for pattern classification and feature/representation learning[31]. Several subcategories of deep learning algorithms exist dependent on the objective of the approaches (figure):

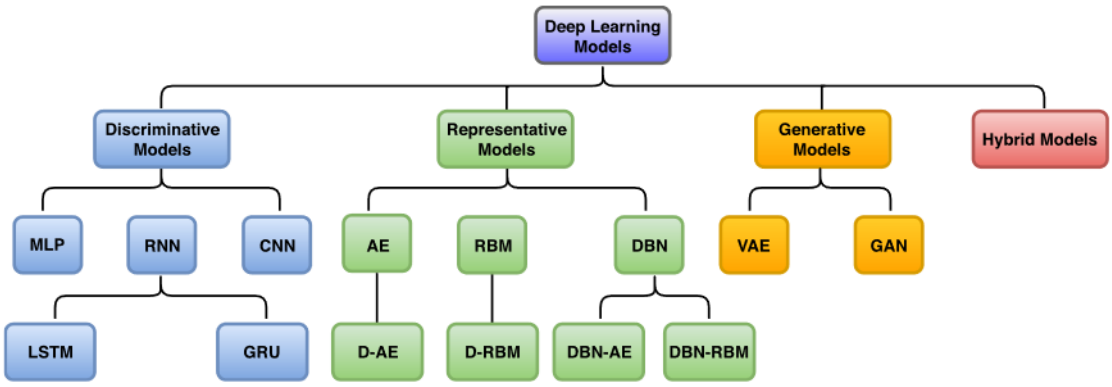
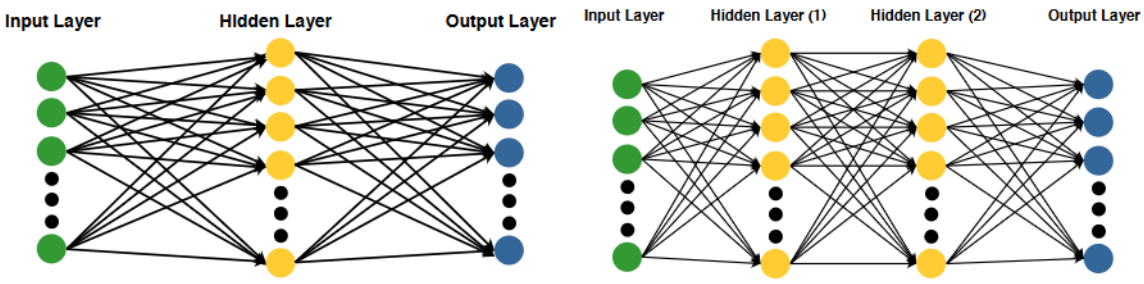


Figure 3.10: Deep learning architectures

Discriminative deep learning models: based on the adaptively learnt discriminative features, which classify the input data into a predefined label. Discriminative algorithms are capable of learning distinctive characteristics by non-linear transformation and classifying via probabilistic prediction9. Consequently, these algorithms can do both feature extraction and classification (corresponding to figure 1). Multi-layer perceptron (MLP) [40], recur- rent neural networks (RNNs) [41], and convolutional neural networks (CNNs) [42], as well as their variants, are the predominant discriminative designs.



(a) Basic fully-connected neural network (b) Multi-Layer Perceptron

Figure 3.11: llustration of standard neural network and multilayer perceptron.

Representative deep learning models: which learn the representative and pure characteristics from the supplied data. These algorithms can only extract features (figure 1), but they cannot perform classification. Autoencoder (AE) [43], restricted Boltzmann machine (RBM) [44], and deep belief networks (DBNs) [45], in addition to their variants, are often employed deep learning techniques for representation.

Generative deep learning models: which discover the joint probability distribution of input data and target label. J. Neural Eng. 18 (2021) 031002 X Zhang et al. are generated primarily by generative algorithms in the brain signal domain. Figure 3. Deep learning models. Based on the algorithm functions, they can be categorized into discriminative, representational, generative, and hybrid models. Multi-layer perceptron (MLP), recurrent neural networks (RNN), and convolutional neural networks (CNN) constitute the majority of discriminative models (appendix B.1) (CNN). Long short-term memory (LSTM) and gated recurrent unit are the two primary components of an RNN (GRU). Representative models (appendix B.2) are authoencoder (AE), restricted Boltzmann machine (RBM), and deep belief networks (DBN). Deep-Autoencoder, or D-AE, refers to an Autoencoder with numerous concealed layers. Similarly, D-RBM refers to a deep-restricted Boltzmann machine with numerous concealed layers. Deep belief network can be composed of either AE or RBM, therefore we separated DBN into DBN-AE and DBN-RBM. Commonly used generative models in non-invasive brain signal analysis include variational autoencoder (VAE) and generative adversarial networks (appendix B.3) (GANs). A collection of brain signal samples used to

augment the training set[15]. Commonly used generative models in brain signal analysis include variational autoencoder (VAE)10 [46], generative adversarial networks (GANs) [47], and others.

Hybrid deep learning models: can incorporate more than two models of deep learning. Typical hybrid deep learning models, for instance, employ representation techniques for feature extraction and discriminative algorithms for classification.

Table 3.1: Summary of deep learning model types.

Deep learning	Output	Function	Training method
Discriminative	Label	Feature extraction Classification	Supervised
Representative	Representation	Feature extraction	Unsupervised
Generative	New Sample	Generation, Reconstruction	Unsupervised
Hybrid	—	—	—

3.5.1 Discriminative Deep Learning Models

Since brain signal detection is the primary function of BCIs, discriminative deep learning models are the most common and effective algorithms. Assume we have a dataset of brain signal samples, X, Y, where X is the set of brain signal observations and Y is the set of sample ground truth values (i.e., labels). Consider the sample-label pair x RN, y RM, where N and M are the number of observations and sample categories, respectively. The objective of discriminative deep learning models is to learn the function x y[15]. Briefly, discriminative models receive input data and return the corresponding category or label. All of the discriminative models shown in this section are supervised learning techniques that require both observation and ground truth information.

Multi-Layer Perceptron (MLP)

Multilayer Perceptron, one of the simplest and most fundamental deep learning models, is derived from the classic neural network's three neuron layers (i.e., an input layer,

a hidden layer, and an output layer). MLP differs significantly from conventional neural networks in that it contains multiple hidden layers. Each node is fully connected to the nodes of the neighbouring layers but not to the nodes of the same layer. MLP contains numerous concealed layers. As illustrated in the preceding figure, we use a structure with two hidden layers to illustrate the data flow in MLP[14]. First, an operation $\tau(x)$ is defined as follows:

$$\tau(x) = w * x + b \tag{3.1}$$

$$\tau(x, x^{'}) = w * x + b + w^{'} * x^{'} + b^{'} \tag{3.2}$$

where x and $x^{'}$ denote two variables while w, $w^{'}$, b, and $b^{'}$ denote the corresponding weights and basis. The input layer receives observation x and forwards it to the initial hidden layer,

$$x^{h1} = \sigma(\tau(x)) \tag{3.3}$$

where x^{h1} represents the data flow in the initial hidden layer and σ is the non-linear activation function. There are various regularly used activation functions, such as sigmoid/Logistic, Tanh, and ReLU. In this part, we will utilize the sigmoid activation function as an example[14]. The data flow then proceeds to the second hidden layer and output layer,

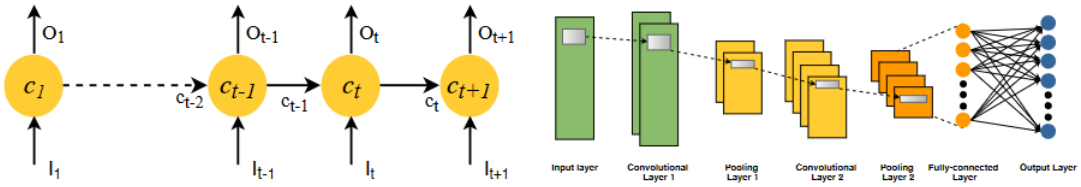
$$x^{h2} = \sigma(\tau(x^{h1})) \tag{3.4}$$

$$x^{'} = \sigma(\tau(x^{h2})) \tag{3.5}$$

$x^{'}$ represents the predicted outcomes in one-hot format. The error (or loss) could be determined based on the distance between $y^{'}$ and y. For instance, the error based on Euclidean distance can be calculated by:

$$error = ||x^{'} - y||_2 \tag{3.6}$$

where $||.||_2$ represents the Euclidean mean The error will then be back-propagated and optimized by an appropriate optimizer. The optimizer will modify the model's weights and basis until the error converges. The most popular loss functions consist of cross-entropy, negative log likelihood, mean square estimation, and others. The most popular optimizers are Adaptive moment estimation (Adam), Stochastic Gradient Descent (SGD), and Adagrad (Adaptive sub-gradient approach), among others. Several terms are readily misconstrued, including Artificial Neural Network (ANN), Deep Neural Network (DNN), and Multilayer Perceptron (MLP)[14]. These concepts have no clear distinction and are frequently interchanged in literature. In general, ANN are neural networks with fewer hidden layers (shallower) than DNN (in this case, DNN is equivalent to MLP).



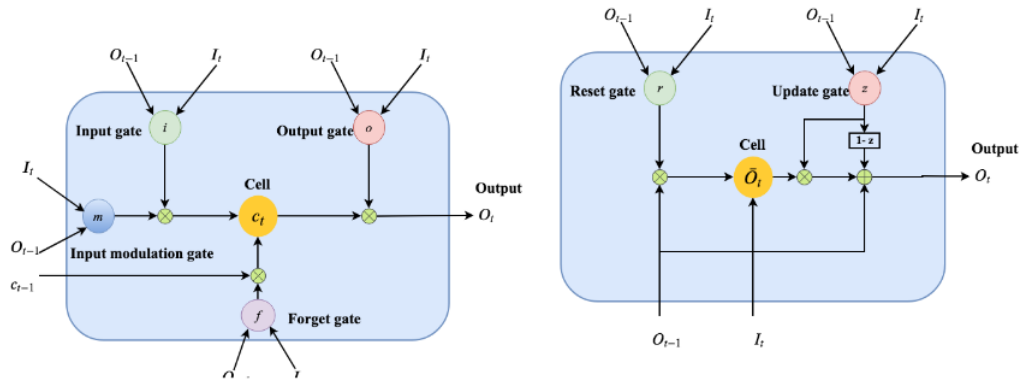
(a) Recurrent Neural Networks (b) Convolutional Neural Networks

Figure 3.12: Illustration of RNN and CNN models.

Recurrent Neural Networks (RNN)

Recurrent Neural Networks are a subclass of discriminative deep learning models designed to capture temporal correlations between input data. The previous figure depicts the time domain activity of a given RNN node. At each time range [1, t + 1], the node receives an input I_t and the prior time's hidden state c. (except the first time). At time t, for example, it receives both the input It and the hidden state of the preceding node c_{t-1}. The hidden state can be thought of as the nodes "memory". which aids the RNN in 'remembering' the historical input. Long short-term memory and gated recurrent units are two exemplary RNN designs that have garnered a considerable deal of interest and enjoyed a great deal of success. They both adhere to the fundamental principles of RNN, and we will focus on the intricate internal structures of each node. We refer to

this structure as a 'cell' because it is far more complex than simple neural nodes[14]. Cells in RNN are equivalent to nodes in MLP.



(a) Structure of LSTM cell (b) Structure of GRU cell

Figure 3.13: Illustration of detailed LSTM and GRU cell structures.

Long Short-Term Memory (LSTM)

At time t, Figure 3.12a depicts the structure of a single LSTM cell. $I_t, O_{t-1}, and C_{t-1}$, Ot1, and ct1 are the three inputs and two outputs of the LSTM cell $C_t and O_{t-1}$. As follows is the procedure:

$$I_t, O_{t-1}, c_{t-1} \rightarrow c_t, O_t \tag{3.7}$$

I_t represents the input value at time t, O_{t-1} represents the output at the preceding time (i.e., time $t-1$), and c_{t-1} represents the hidden state at the preceding time. c_t and O_t signify the concealed state and output at time t, respectively. Therefore, we can observe that the output O_tt at time t is tied not just to the input I_t, but also to the information from the prior time. Thus, LSTM is enabled to recall pertinent information in the temporal domain[15]. In addition, the primary purpose of LSTM is to regulate the memory of specific information. In order to accomplish this, the LSTM cell employs four gates: the input gate, the forget gate, the output gate, and the input modulation gate. Each gate is a weight that controls the amount of information that can pass through it. For instance, if the weight of the forget gate is zero, the LSTM cell will recall all the information passed since time $t-1$; if the weight is one, the LSTM cell will forget everything. The weight is determined by the activation function that

corresponds to it. The specific data flow looks as follows:

$$f = \sigma(\tau(I_t, O_{t-1})) \tag{3.8}$$

$$i = \sigma(\tau(I_t, O_{t-1})) \tag{3.9}$$

$$o = \sigma(\tau(I_t, O_{t-1})) \tag{3.10}$$

$$m = \sigma(tanh(I_t, O_{t-1})) \tag{3.11}$$

$$c_t = f * c_{t-1} + i * m \tag{3.12}$$

$$h_t = o * tanh(c_t) \tag{3.13}$$

Gated Recurrent Units (GRU)

GRU is another popular RNN design. Similarly to LSTM, GRU aims to exploit historical information. GRU does not require hidden states, but it only receives temporal data from the output of time $t-1$. Figure 3.13b demonstrates that GRU has two inputs (I_t and O_{t-1}) and one output (O_t). The mapping is described as follows:

$$I_t, O_t \rightarrow O_t \tag{3.14}$$

Two gates make up GRU: reset gate r and update gate z. The former determines how the input is combined with past memories. Comparable to the forget gate of LSTM, the latter determines how much of past memory to retain. The flow of data is as follows:

$$z = \sigma(\tau(I_t, O_{t-1})) \tag{3.15}$$

$$r = \sigma(\tau(I_t, O_{t-1}) \tag{3.16}$$

$$\bar{O}_t = tanh(\tau(I_t, r * O_{t-1}) \tag{3.17}$$

$$O_t = (1-z) * O_{t-1} + z * \bar{O}_t \tag{3.18}$$

Observation reveals that the intermediate variable $\bar{O}_t$ is comparable to the hidden state of LSTM. However, $\bar{O}_t$ only functions at this time point and cannot pass to the following time point. We provide a brief comparison of LSTM and GRU given their similarities. According to research, LSTM and GRU have equivalent performance. For every specific task, it is recommended to compare the performance of two options to determine which one is superior. Second, GRU is a lightweight algorithm with only two gates and no hidden state. GRU is hence easier to train and requires fewer training examples for generalization. Third, LSTM generally performs better if the training dataset is sufficiently large.

Convolutional Neural Networks (CNN)

Convolutional Neural Networks are one of the most prevalent deep learning models specializing in the investigation of spatial data. This section will provide a basic overview of CNN's operational structure. CNN is frequently utilized to uncover the latent spatial information in applications such as image recognition, ubiquity, and object search- ing because to its prominent characteristics, including regularized structure, good spatial localization, and translation invariance. CNN is intended to capture the differential interdependence between the patterns associated with various brain signals in BCI. The conventional CNN architecture is seen in Figure 3.12b. One input layer, two convolutional layers, each followed by a pooling layer, one fully-connected layer, and one output layer comprise the CNN[15]. Each layer's square patch indicates the processing status of a batch of input values. The objective of the CNN is to transform the input data into a form that is simpler to recognize with minimal information loss. CNN is composed of three layers: the convolutional Layer, the pooling Layer, and the fully-connected Layer. The convolutional layer comprises a set of filters to convolute the input data, followed by a nonlinear transformation to extract geographical characteristics[15]. Several essential hyper-parameters, such as the number of filters and the size of each filter, must be configured in the convolutional layer of the deep learning implementation. The pooling layer follows the convolutional layer in most cases.

The objective of the pooling layer is to gradually lower the spatial size of the features. Thus, it is possible to reduce the number of parameters (e.g., weights and base) and the computational overhead. Three types of pooling operations exist: maximum, minimum, and average. Take maximum pooling for instance. The procedure of pooling yields the maximum value of the pooling area. The pooling layer's hyper-parameters consist of the pooling operation, pooling area size, strides, etc. In the fully-connected layer, as in the fundamental neural network, the nodes have complete connections to all activations in the layer underneath them.

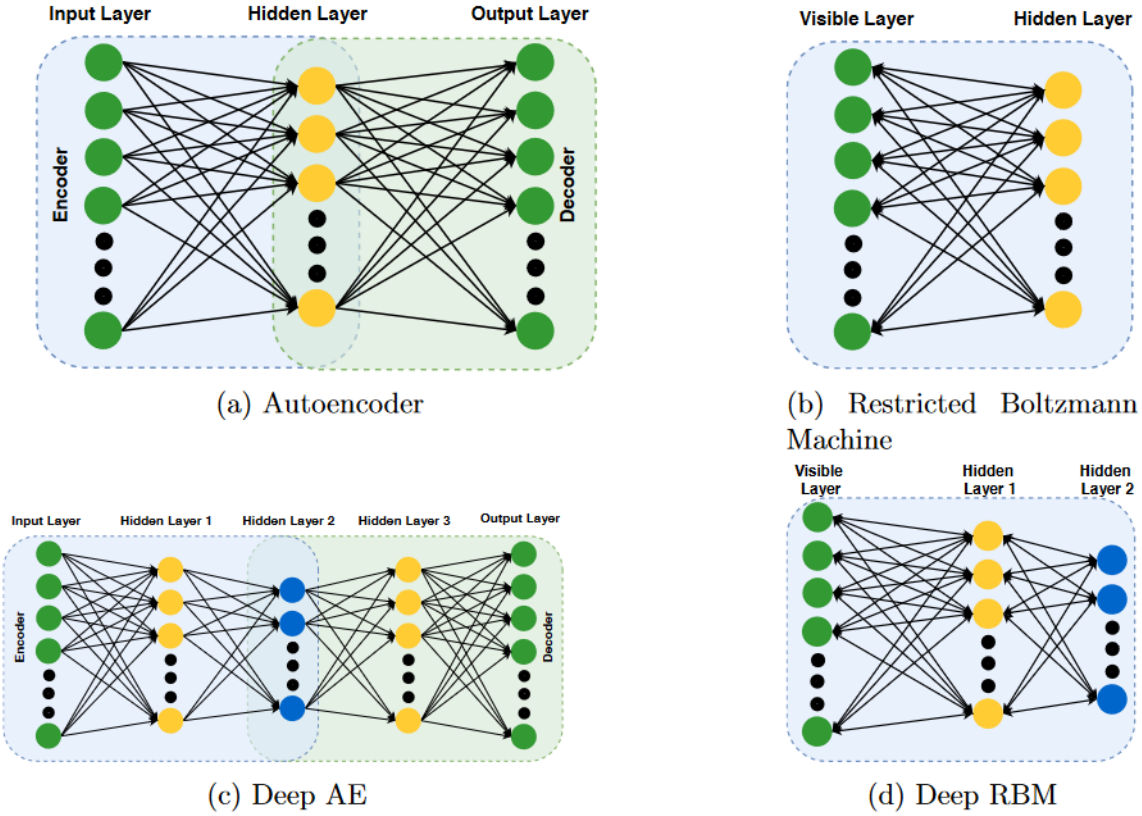
3.5.2 Representative Deep Learning Models

Autoencoders and limited Boltzmann machines are the building elements of representative deep learning models. Deep Belief Networks consist of either AE or RBM. Unsupervised learning approaches comprise the representative models AE, RBM, and DBN. Consequently, they can learn the representative characteristics using only the input observations x and not the ground truth y. Briefly, representative models receive the input data and generate a dense representation of the data. Several models (such as DBN, Deep RBM, and Deep AE) have multiple definitions in different research; we have selected the most understandable definitions and will discuss them in depth in this part.

Autoencoder (AE)

An autoencoder is a neural network with three layers, as shown in Figure 3.14a: the input layer, the hidden layer, and the output layer. The AE is trained to rebuild its inputs, which forces the hidden layer to attempt to learn accurate representations of the inputs. AE's structure consists of two blocks. The first block is known as the encoder, and it encodes the observation into a latent representation (sometimes known as a "code").

$$x^h = \sigma(\tau(x)) \tag{3.19}$$



(a) Autoencoder

(b) Restricted Boltzmann Machine

(c) Deep AE

(d) Deep RBM

Figure 3.14: Illustration of standard representative models for deep learning. a) A fundamental autoencoder has three layers, with the input layer and output layer having identical values. Encoding occurs from the input layer to the hidden layer, whereas decoding occurs from the hidden layer to the output layer. b) In the Restricted Boltzmann Machine, the transformation weights of the encoder and decoder are identical. The input and output layers are combined to form the visible layer. (c) The stacked autoencoder conceals multiple layers. Typically, there are an odd number of concealed layers, and the intermediate layer contains learnt representative features[15]. d) The deep RBM consists of one visible layer and numerous hidden levels, with the last layer containing the encoded representation.

where x^h is the layer that is hidden. The second block is known as the decoder since it decodes the representation into the original space.

$$y^{'} = \sigma(\tau(x^h)) \tag{3.20}$$

where $y^{'}$ represents the output. AE equalizes $y^{'}$ to the input x and determines the error based on the distance between them. Consequently, AE can only compute the loss

function using x, not the ground truth y.

$$error = ||y^{'} - x||_2 \tag{3.21}$$

Unlike Equation 3.6, this equation does not include the variable y because it uses the input x as the starting point. This explains why AE is capable of unsupervised learning. Obviously, one type of AE is Deep-AE (D-AE), which has multiple hidden layers. Figure 3.14c depicts the construction of D-AE with three hidden layers. From the illustration, we can see that both the encoder and the decoder have an additional hidden layer[15]. The symmetry of the structure guarantees the efficiency of the encoding and decoding processes. Thus, D-AE typically has an odd number of hidden layers (e.g., $2n+1$), where the first n layers belong to the encoder, the $(n+1)-th$ layer serves as the shared code between the encoder and decoder, and the last n layers belong to the decoder. The data flow of D-AE can be represented as (Figure 3.14c):

$$x^{h1} = \sigma(\tau(x)) \tag{3.22}$$

$$x^{h2} = \sigma(\tau(x_{h2})) \tag{3.23}$$

x^{h2} represents the middle hidden layer (the code). Then we may decode the hidden layer to obtain

$$x^{h3} = \sigma(\tau(x_{h2})) \tag{3.24}$$

$$y^{'} = \sigma(\tau(x_{h3})) \tag{3.25}$$

D-AE is nearly identical to AE except that it contains more hidden layers. AE has numerous versions than D-AE, including denoising autoencoder, sparse autoencoder, contractive AE, etc. Because the D-AE is readily mistaken with the AE-based deep belief network, we simply introduce it here.

The basic principle behind AE and its derivatives is straightforward: condense the input data x into a code x^h (the code layer often has smaller dimension) and then recreate the data using the code. If the reconstructed $y^{'}$ can resemble the input data x, it

can be shown that the condensed code x^h has sufficient information about x; therefore, x^h can be regarded as a representation of the input data for future operations (e.g., classification)

Restricted Boltzmann Machine (RBM)

Restricted Boltzmann Machine is a neural network that can learn a probability distribution across its inputs. As shown in Figure 3.14b, it consists of two layers: one visible layer (input layer) and one buried layer. The connection lines between the two levels are bidirectional, as shown in the diagram. RBM is a variation of the Boltzmann Machine that has more stringent restrictions regarding the absence of intra-layer connections. Similarly to AE, the RBM technique consists of two phases. The initial stage compresses the input data from the original space to the latent space's hidden layer[15]. The hidden layer is then utilized to reconstruct the input data identically. RBM has a stricter constraint than AE, which is that the encoder and decoder weights must be identical. We have

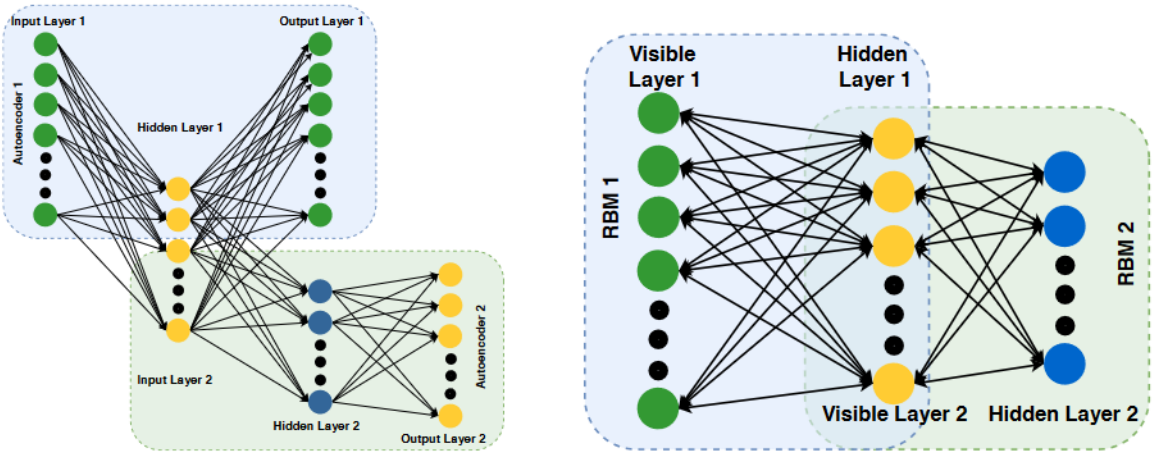
$$x^h = \sigma(\tau(x)) \tag{3.26}$$

$$x^{'} = \sigma(\tau(x_h)) \tag{3.27}$$

In the two equations presented above, the weights of $\tau(.)$ are identical. The training error can then be estimated by:

$$error = ||x^{'} - x||_2 \tag{3.28}$$

Figure 3.14d reveals that the Deep-RBM (D-RBM) is an RBM with numerous concealed layers. The visible layer's input data travel first to the first hidden layer and then to the second hidden layer. The code will then flow in reverse into the visible layer for reconstruction.



(a) DBN-AE (b) DBN-RBM

Figure 3.15: Exemplification of extensive belief networks A DBN made up of autoencoders. DBN-AE consists of numerous AE components (in this case, two AE), with the hidden layer of the preceding AE functioning as the input layer of the following AE. The last AE's hidden layer is the learned representation. (b) DBN made up of RBM. The first RBM's hidden layer functioning as the visible layer of the second RBM. The final hidden layer is the representation code. While DBN-RBM and D-RBM have a similar architecture (Figure 3.14d), the former is trained greedily while the latter is trained cooperatively.

Deep Belief Networks (DBN)

A Deep Belief Network (DBN) is a stack of simple networks including AEs and RBMs [53]. Thus, we separated DBN into DBN-AE (also referred to as stacked AE) which consists of AE and DBN-RBM (also referred to as stacked RBM) which consists of RBM.

As depicted in Figure 3.15a, the DBN-AE is comprised of two AE structures, with the hidden layer of the first AE functioning as the input layer of the second AE. This graphic consists of two steps. In the initial stage, the input data fed to the initial AE adhere to the preceding principles[15]. Calculating and back propagating the reconstruction error to modify the weights and base. This cycle is repeated until the AE converges. We acquire the mapping:

$$x^1 \rightarrow x^{h1} \tag{3.29}$$

Then, we proceed to the second stage, where the learned representative code in the

hidden layer x^{h1} is utilized as the input layer of the second AE, which is:

$$x^2 = x^{h1} \tag{3.30}$$

and then, after the second AE converges, we have

$$x^2 = x^{h2} \tag{3.31}$$

x^{h2} represents the hidden layer of the second AE, whereas xh3 represents the ultimate result of the DBN-AE. AE is based on the concept of learning a representative code with lower dimensionality that contains the majority of the input data's information. The purpose of DBN-AE is to acquire a more pure and representational code.

Likewise, the DBN-RBM is formed by a number of individual RBM structures. Figure 3.15b depicts a DBN comprised of two RBMs in which the hidden layer of the first RBM serves as the visible layer of the second RBM.

Compare the DBN-RBM (Figure 3.15b) with D-RBM (Figure 3.15a) (Figure 3.14d). They nearly share the same architectural design. Moreover, the architectures of DBN-AE (Figure 3.15a) and D-AE (Figure 3.14c) are comparable. The most significant distinction between the DBN and deep AE/RBM is that the former is trained independently while the latter is developed collaboratively. In specifically, the first AE/RBM is trained initially for the DBN, and when it converges, the second AE/RBM is trained[54]. For the deep AE/RBM, jointly training entails training the entire structure, regardless of how many levels it contains.

3.5.3 Generative Deep Learning Models

The primary purpose of generative deep learning models is to produce training samples or augment data. In other words, generative deep learning models benefit the BCI sector by improving the quality and amount of training data. After augmenting the data, discriminative models will be used for classification. This method is designed to improve the robustness and efficacy of trained deep learning networks, particularly

when training data is limited. In brief, the generative models receive the input data and generate a set of similar output data. In this part, we will introduce two standard deep learning generative models: VAE and GAN.

Variational Autoencoder (VAE)

Initiated in 2013 [55], Variational Autoencoder is a significant version of AE and one of the most potent generative algorithms. The standard AE and its variants can be utilized for representation, but cannot be used for generation because the learned code (or representation) may not be continuous. Therefore, it is impossible to generate a random sample that resembles the input sample. In other words, interpolation is not supported by the standard AE. Thus, the input sample can be replicated, but a similar sample cannot be generated. This trait is what makes VAE so valuable for generative modeling: the latent spaces are meant to be continuous, which makes random sampling and interpolation straightforward[15]. Next, we shall explain how VAE operates.

Similar to regular AE, VAE can be divided into an encoder and decoder, with the encoder embedding input data into a latent space and the decoder transferring data from the latent space to the original space. Nevertheless, the learned representation in the latent space is constrained to approximate a prior distribution $p(\bar{z})$, which is typically set to Standard Gaussian distribution. Based on the re-parameterization approach [55], the first hidden layer of VAE is composed of two portions, one of which represents the expectation μ and the other the standard deviation σ. Consequently, we have:

$$\mu = \sigma(\tau(x)) \tag{3.32}$$

$$\sigma = \sigma(\tau(x)) \tag{3.33}$$

The latent coding in the hidden layer is then sampled from a Gaussian distribution $N(\mu, \sigma^2)$ rather than being directly calculated. Statistical code:

$$z = \mu + \sigma * \varepsilon \tag{3.34}$$

where $\varepsilon \sim N(0,I)$ The representation z is constrained to a prior distribution, and Kull-Back Leibler divergence is used to estimate the distance $error_{KL}$.

$$error_{KL} = D_{KL}(z, p\bar{(z)}) \tag{3.35}$$

where $p\bar{(z)}$ is the preceding distribution. The decoder converts z into the output $y^{'}$,

$$y^{'} = \sigma(\tau(z)) \tag{3.36}$$

and the reconstruction error is

$$error_{recon} = ||y^{'} - x||_2 \tag{3.37}$$

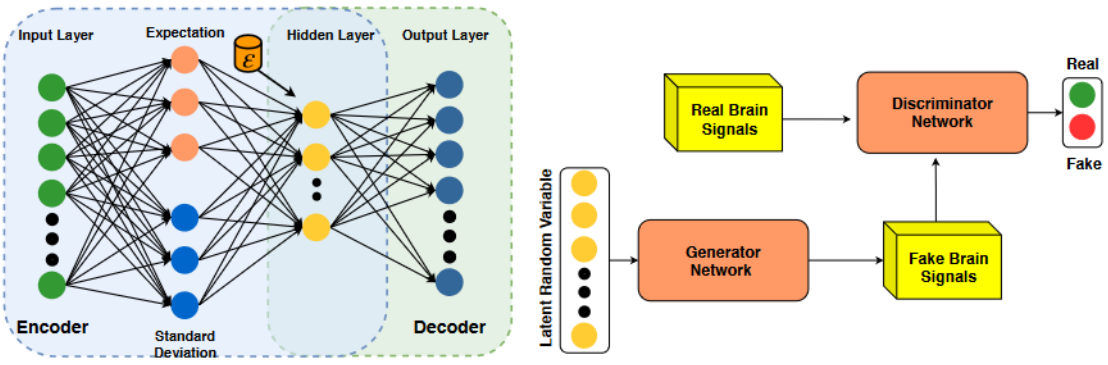
The overall error for VAE is combined by the DL divergence and the reconstruction error,

$$error = error_{KL} + error_{recon} \tag{3.38}$$

The most important aspect of VAE is that all latent representations z must adhere to the normal distribution. Thus, it is possible to randomly pick a representation $z^{'} \in p\bar{(z)}$ from the previous distribution and then reconstruct a sample using $z^{'}$. This is the reason VAE is so effective in generation.

Generative Adversarial Networks (GAN)

The Generative Adversarial Networks [56] paper was published in 2014 and has been a huge success in a variety of academic fields (e.g., computer vision and natural language processing). GAN consists of two neural networks containing a generator and a discriminator that are simultaneously trained. The generator captures the distribution of the input data, while the discriminator estimates the likelihood that a sample was drawn from the training data. The generator strives to produce bogus samples, whereas the discriminator seeks to determine whether the sample is authentic. Because the functions of the generator and discriminator are opposed, GAN is referred



(a) Variational Autoencoder (b) Generative Adversarial Networks

Figure 3.16: Illustration of deep learning generative models. (a) VAE has two concealed layers. The first hidden layer consists of two components that are learned independently from the input layer: the expectation and the standard deviation. The encoded information is represented by the second hidden layer. represents the normal distribution standard. b) GAN primarily consists of the generator network and the discriminator network. The former gets a latent random variable to generate a fictitious brain signal, whereas the latter receives both the real and generated brain signals and attempts to discern whether the signal was generated or not. Instead of categorization, GAN reconstructs or augments data in BCI.

regarded as adversarial. After both the generator and discriminator have converged, the discriminator should be unable to recognize the generated samples. Thus, the pre-trained generator can be used to generate a batch of samples for classification and other activities[14].

Figure 3.16b depicts the conventional GAN technique. The generator receives a noise signal s sampled at random from a multimodal Gaussian distribution and outputs simulated brain signals x_F. The discriminator receives genuine brain signals x_R and a created fake sample x_F, and then guesses whether the sample received is real or phony[14]. The generator and discriminator's internal architectures are created based on the data kinds and circumstances. CNN's exceptional ability to extract spatial characteristics allows us, for instance, to construct GAN using convolutional layers on fMRI pictures. Together, the discriminator and the generator are trained. After convergence, the generator can generate many brain signals x_G. To train a more effective and resilient classifier, the training set is expanded from x_R to $\{x_R, x_G\}$.

3.5.4 Hybrid Model

Hybrid deep learning models are models comprised of at least two deep basic learning models, where the basic model is either a discriminative, representative, or generative deep learning model. Based on their objectives, hybrid models are divided into two subcategories: classification-aimed (CA) hybrid models and non-classification-aimed (NCA) hybrid models[14].

The majority of deep learning-related BCI investigations concentrate on the first group. The representative and generative models are employed to improve the discriminative models based on the available literature. The representative models can provide more informative and low-dimensional features for discriminating, whilst the generative models can help improve the quality and quantity of training data, hence providing more information for classification. The CA hybrid models can be further subdivided as follows: 1) several discriminative models combined to extract more distinct and robust features (e.g., CNN+RNN); 2) a representative model followed by a discriminative model (e.g., DBN+MLP); 3) a generative model combined with a representative model followed by a discriminative model; and 4) a generative model combined with a representative model followed by a non-deep learning classifier. A few NCA hybrid models seek to reconstruct brain signals. St-yves et al. [57] used GAN to recreate visual stimuli from fMRI pictures, for instance[14].

3.6 Conclusion

In this chapter, we provide a comprehensive overview of current breakthroughs in deep learning and machine learning as they pertain to brain-computer interface applications. Deep learning allows us to acquire high-level features automatically from BCI signals and relies less on manually-crafted features and domain knowledge than conventional methods. We present a summary of the most prevalent deep learning models.

Chapter 4

Experimental implementation and results

4.1 Introduction

In this chapter, we will describe the algorithms, datasets, hardware, and software utilized to produce the following findings. In addition, we detail the many technical aspects associated with the deployment and implementation of our system. We discuss the dataset collection from several local subjects and the conditions while collecting it and the used EEG headset. Then, we discuss our proposed model and its flowchart, detailing each part of it and the algorithms that we have used (preprocessing, artifact removing, feature extraction, classification, etc.). Finally, we will highlight some robotic notations in order to facilitate our process.

4.2 Development software and hardware

4.2.1 EEG Headset

As discussed previously, there are two methods for acquiring EEG signals: invasive and noninvasive. We concentrate on the noninvasive technique on which all EEG-based brain-controlled robotics examples are built. In accordance with the worldwide 10–20 standard, raw electrical signals are captured by sensors (electrodes) placed on the scalp in this procedure. The spacing between adjacent pairs of electrodes in this standard system is either 10 or 20 percent of the diameter of the scalp. In our study, we used Emotiv EPOC+, it is a wireless, multi-channel, high-resolution neuroheadset. The EPOC utilizes a set of 14 sensors and two references to tune into electric impulses produced by the brain in order to detect the user's thoughts, emotions, and facial expressions in real time with a sampling rate of 128 Hz, that registers brain signals (in

μV) through different softwares, in this project we used the EMOTIV-PRO app, it comes with several features, however we only focused on the registration, labeling and data exportation. The EPOC wirelessly connects to the PC.

Figure 4.1: Framework of MI-BCI control system

4.2.2 EEG Electrodes Gel

Electrode Gel is a highly conductive saline gel that has been rigorously tested and is highly recommended for Neuroelectrics studies needing gel. This comprises tests and interventions including the use of NG Geltrodes and Foretrodes for EEG monitoring and NG Pistims for hybrid tES/EEG applications. The gel is utilized to transport an electrical current from the skin to a measurement device. A sticky patch may be used to adhere the gel electrode to the skin in order to measure the electrical activity of the heart or brain. In this study, i used Bio True gel (BAUSCH + LOMB).

4.2.3 EEG Mice Software

EEG mice have a graphical user interface built by me with the NeuroScience Laboratory at Sapienza University. Its main role is to preprocess EEG signals; cream extrac-

Figure 4.2: The used Gel Electrodes

tion, search by time, search by epochs, filtering, etc. EEG mice have been built using the Tkinter Library in Python and several novel algorithms designed for preprocessing EEG. It has been widely used in this study, especially in preprocessing steps.

4.2.4 Training Hardware

For the preprocessing and model training, i have used my own laptop with the following characteristics:

1. **CPU:** AMD Ryzen 7 4800H @2.90 GHz.
2. **GPU:** Gtx 1660ti.
3. **RAM:** 16.00 GB DDR4
4. **Storage:** 512GB SSD, 1.00TB HDD

4.2.5 Languages and FrameWorks

Python: Python is a high-level, interpretable, object-oriented programming language with dynamic semantics [2021]. Its high-level data structures, dynamic typing, and vast library collection make it the ideal language for AI and IoT programming.

OpenCV: OpenCv is a real-time optimized Computer Vision library, tools, and hardware that was created by Intel.

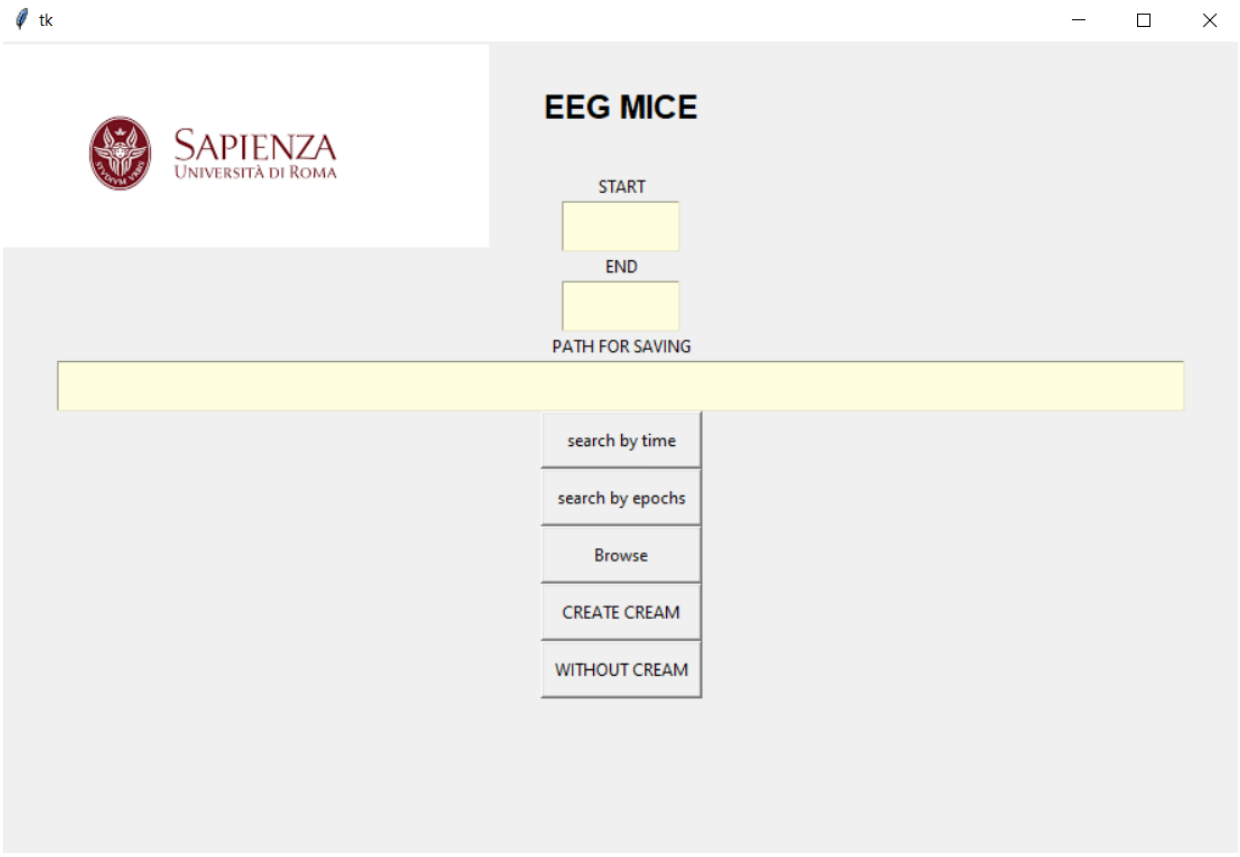


Figure 4.3: GUI of eeg mice for EEG signal preprocessing

Pycharm: The coding platform is ”pycharm,” an integrated development environment (IDE) used in computer programming, specifically for the Python language. It is being developed by JetBrains, a Czech business. It provides code analysis, a graphical debugger, an integrated unit tester, version control system (VCS) integration, and support for web development with Django and data science with Anaconda.

Matplotlib: Matlplotlib is a plotting library for the Python computer language and NumPy, its extension for numerical mathematics. It provides an object-oriented API for embedding charts into applications utilizing GUI toolkits such as Tkinter, wxPython, Qt, and GTK+.

Tensorflow: TensorFlow is an open-source machine learning library. It can be applied to a variety of tasks, but its primary focus is on the training and inference of deep neural networks and machine learning.

PyTorch: PyTorch is an open source machine learning framework based on the Torch library that is used for applications such as computer vision and natural language processing. It was developed primarily by Meta AI. It is open-source, free software distributed under the Modified BSD license.

Keras: The open-source library Keras provides a Python interface for artificial neural networks. Keras serves as the TensorFlow library's interface. Keras supported many backends prior to version 2.3, including TensorFlow, Microsoft Cognitive Toolkit, Theano, and PlaidML.

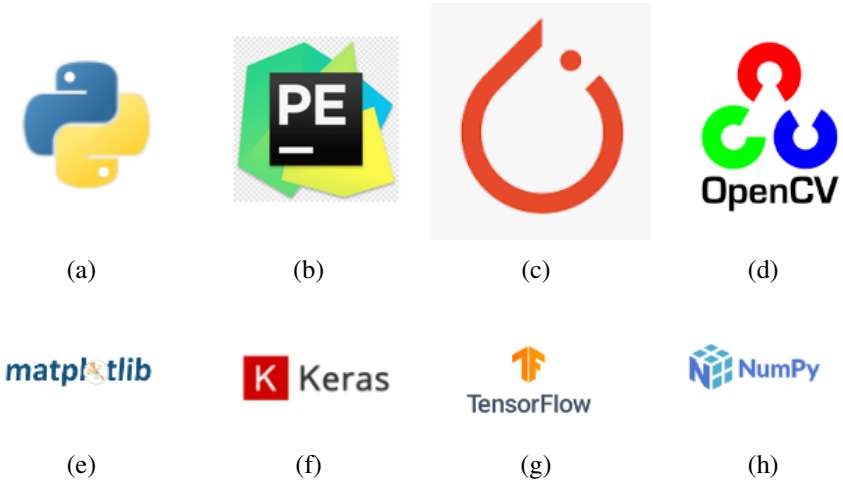
Numpy: NumPy is a Python library that adds support for massive, multidimensional arrays and matrices, as well as a vast number of high-level mathematical functions to operate on these arrays.

MNE Library: MNE-Python is an open-source Python tool for exploring, displaying, and interpreting human neurophysiological data, including MEG, EEG, sEEG, and ECoG, among others.

Jupyter NoteBook: The mission of Project Jupyter is to "create open-source software, open-standards, and services for interactive computing in dozens of programming languages." It was derived from IPython by Fernando Pérez and Brian Granger in 2014

4.3 Proposed Model

Exogenous and endogenous brain-computer interface systems are the two types. The Exogenous brain-machine interface must employ exterior inputs to induce precise brain responses. Electroencephalogram (EEG) evoked patterns of brain-machine interfaces generally consist of the event-related potential P300[196] and steady-state visual evoked potentials (SSVEP)[197]. Endogenous brain-computer interfaces rely on the brain's ability to maintain its own rhythm and do not require external stimulation.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 4.4: (a) Python Logo (b) Pycharm Logo (c) Pytorch Logo (d) OpenCV Logo (e) Matplotlib logo (f) Keras Logo (g) Tensorflow logo (h) numpy logo

They are intrinsically linked to human motion intent and can more precisely reflect the subject's autonomous intent. The most commonly employed paradigm for external BCIs is motor imagery (MI), also known as active BCIs. Exogenous BCIs have a stable signal, require less time for specialized training, and are easy to configure, making the system adaptable to a wide spectrum of persons. However, because exogenous BCIs are not directly controlled by the user, not only are they dependent on external stimuli, but they also require the user's attention, which is likely to cause fatigue[1]. A full EEG-based BCI system consists of four components: EEG data acquisition, signal preprocessing, feature extraction, and pattern classification[**?**]. As depicted in the image below, we suggest a multi-method approach that simultaneously employs SVM and attention-based Bi-LSTM in the final phase to control robots. Later, the controller will receive the most accurate prediction across a given timestamp.

4.4 Data Set

The employed data set in this study is our collected data-set and its currently under licensing. Several 14-channel EEG trials acquired from 8 male drug addicted patients, 6 male healthy patients, and 3 alcoholic patients. Six patients were under drug effects and caffeine effects in the last 12h before the experiments. The mean age of patients was 24.5 year. The first part of trials is paced breathing while the patients are asked to

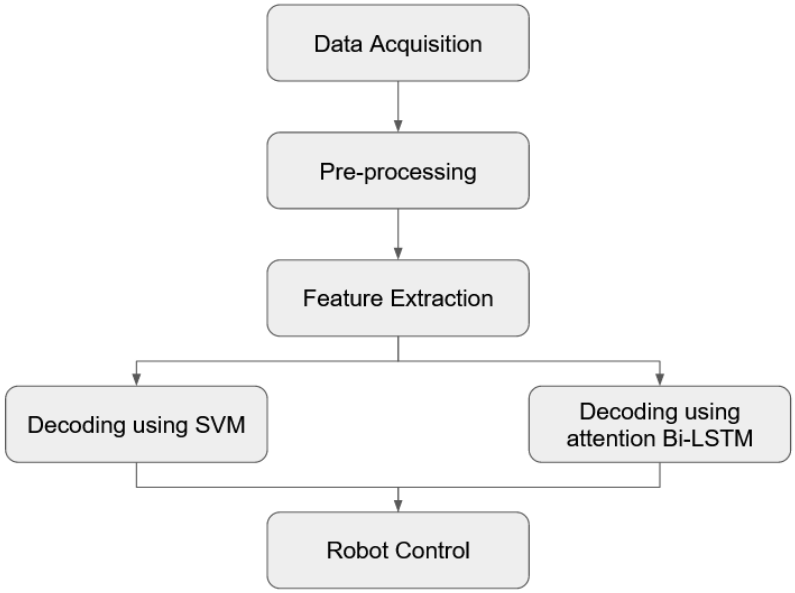


Figure 4.5: Framework of MI-BCI control system

relax and breath in order to use it as a rest reference. The second part is performing four motor imagery (MI) tasks i.e., imagining right-hand fist (RH), imagining left-hand fist (LH), imagining right-leg fist(RL), imagining left-leg fist(LL). Each trial is a 16-seconds experiment with only one single task and 128 Hz sample rate; in total 16 trials for each task has been recorded. The timestamps for each trial is 2040 time stamp. After explaining the constraints, requirements, and potential uses of the trials, all volunteers gave their written agreement.

4.5 System Overview

4.5.1 Support Vector Machine

Data Preprocessing: The raw EEG data was collected with the EMOTIV EPOC+ headset, which comprises 14 electrodes whose placements do not correspond perfectly to those of the standard 10–20 system. According to Pfurtscheller[198][42], due to the fact that the C3 and C4 locations can encompass a portion of the motor cortex, they are the best sites for exploiting Event Related De-synchronization (ERD) and Event Related Synchronization (ERS). In order for F3 and F4 to cover the positions of

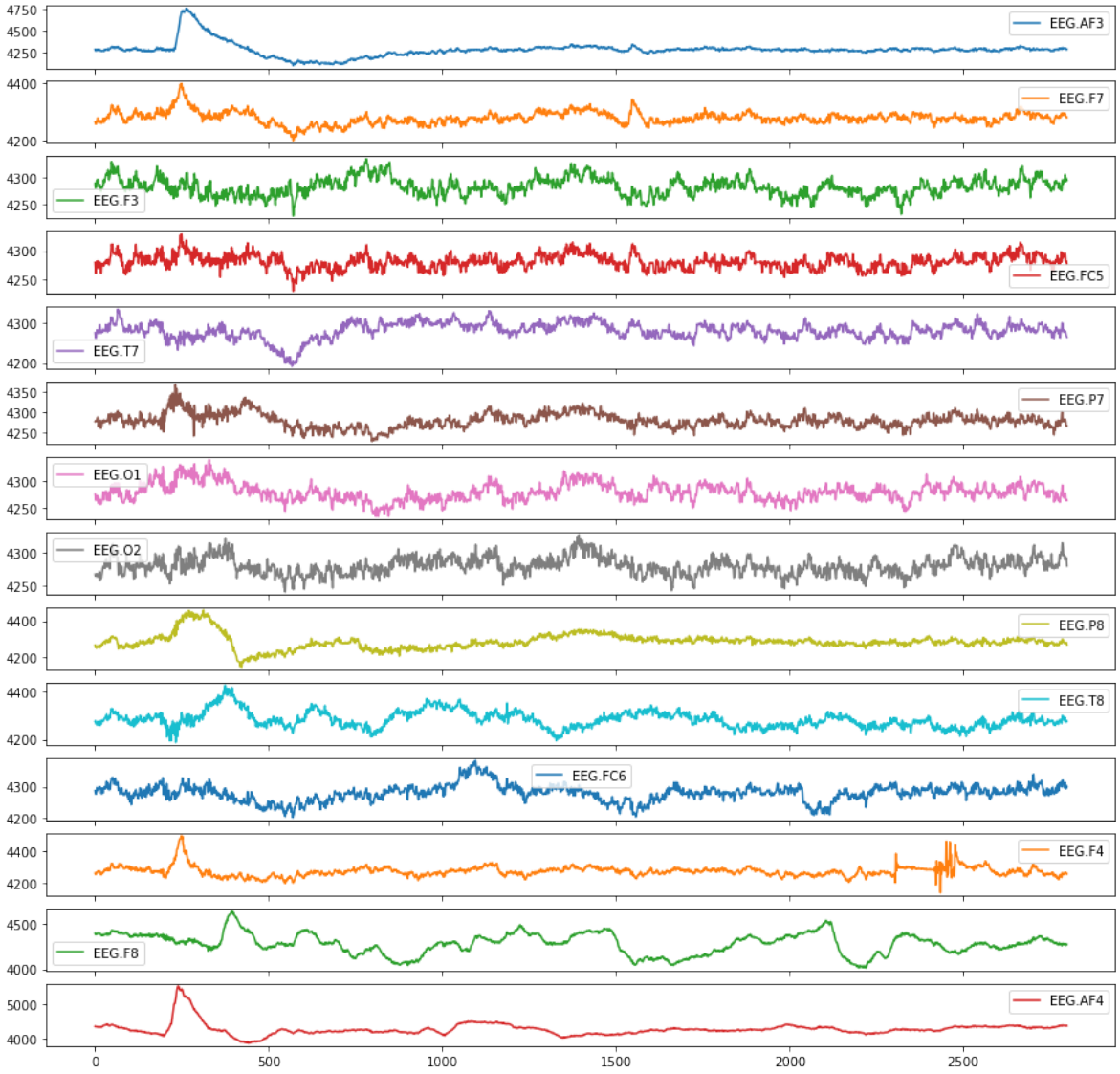


Figure 4.6: Data visualization of sample eeg raw.

C3 and C4, the headset was angled along the axis of the reference electrodes located behind the ears[199]. In addition, we utilized AF3 and AF4 electrodes as well as two reference electrodes to eliminate EMG and EOG artifacts. Then, we employed a high-pass filter to eliminate frequencies below 0.1 or 1 Hz and a low-pass filter to eliminate frequencies above 40 or 50 Hz.

High/low-pass filters and notch When analyzing the measurement of a hole, large variations are crucial. To eliminate these gradual changes unrelated to the brain, we apply some filtering. The same method is used for frequencies above those emitted by the brain. This is the whole spectrogram of each channel's raw data. It is carried out

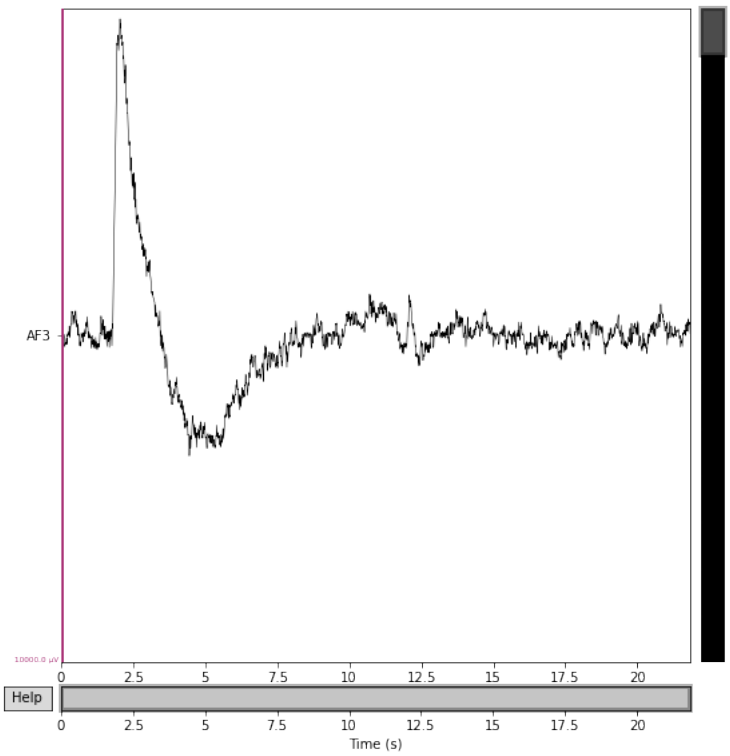


Figure 4.7: Data visualization of AF3 electrode data.

utilizing Welch's approach.

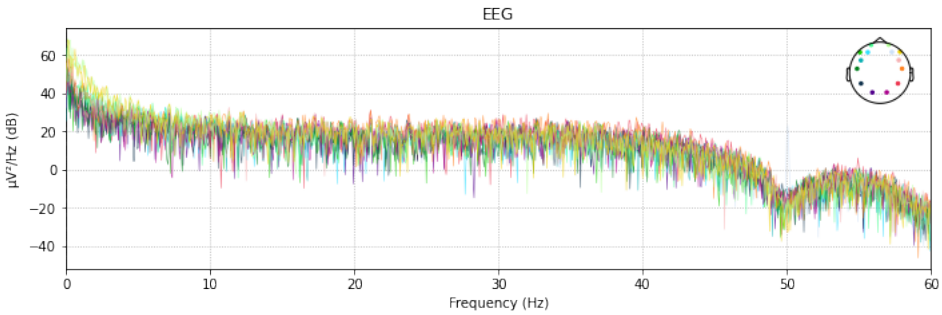


Figure 4.8: Data before applying filters.

Additionally, there are set frequencies associated with electromagnetic noise. For example, the extremely intense peak at 50Hz (and its harmonic at 100Hz) corresponds to power-line diploe emission.

This is filtered with a notch filter.

Since we've made sure of good electrode placement while collecting the dataset, we didn't remove "bad electrode data" and filter it. Because of The brain is not a unique source of electrical activity in the body. Hearbeat, muscle contraction, and

```
Filtering raw data in 1 contiguous segment
Setting up band-pass filter from 0.5 - 50 Hz

FIR filter parameters
---------------------
Designing a one-pass, zero-phase, non-causal bandpass filter:
- Windowed time-domain design (firwin) method
- Hamming window with 0.0194 passband ripple and 53 dB stopband attenuation
- Lower passband edge: 0.50
- Lower transition bandwidth: 0.50 Hz (-6 dB cutoff frequency: 0.25 Hz)
- Upper passband edge: 50.00 Hz
- Upper transition bandwidth: 12.50 Hz (-6 dB cutoff frequency: 56.25 Hz)
- Filter length: 845 samples (6.602 sec)

Effective window size : 16.000 (s)
```

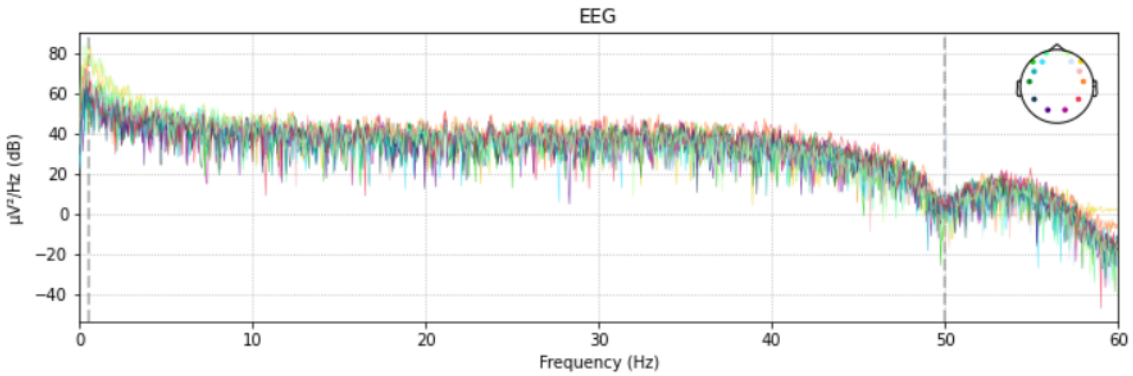


Figure 4.9: Data After applying high/low-pass filters with filtering parameters.

others introduce peridoic noise. These activities can be recorded simultaneously and can be used to remove their induced artifacts. In our case, we did not apply this step, and that is because we chose only 4 electrodes to work with. The EOG artifact has been removed manually with matlab scripts using differential potentials between each pair of electrodes. Finally, we visualize wave bands in the following figure.

Feature Extraction: The Fourier transform method was used with Principal Component Analysis to reduce the dimensionality of the retrieved features when analyzing EEG data. The frequency ranges [8 Hz–12 Hz] and [12 Hz–22 Hz] are responsible for sensorimotor rhythms that develop when a person makes or imagines a movement[200]. We were especially concerned with frequency components between 8 and 22 hertz. Principal component analysis is a common statistical technique for feature extraction and dimensionality reduction. Principal component analysis (PCA) is a linear projection that reduces a set of potentially correlated variables to a smaller set of uncorrelated variables known as principal components. Each next principle component

```
Designing a one-pass, zero-phase, non-causal bandstop filter:
- Windowed time-domain design (firwin) method
- Hamming window with 0.0194 passband ripple and 53 dB stopband attenuation
- Lower passband edge: 49.38
- Lower transition bandwidth: 0.50 Hz (-6 dB cutoff frequency: 49.12 Hz)
- Upper passband edge: 50.62 Hz
- Upper transition bandwidth: 0.50 Hz (-6 dB cutoff frequency: 50.88 Hz)
- Filter length: 845 samples (6.602 sec)

Effective window size : 16.000 (s)
```

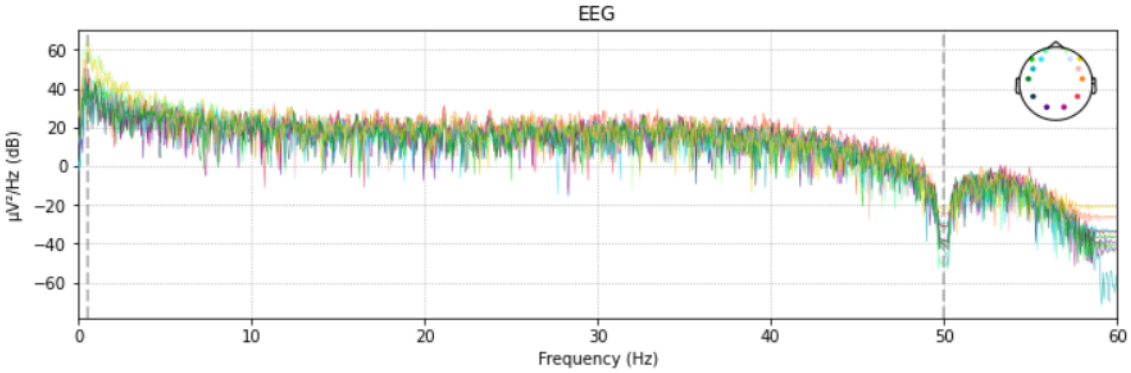


Figure 4.10: Data After applying notch filter with filtering parameters.

has the highest variance and is orthogonal to the existing components. Karl Pearson initially created PCA in 1901. Either the Covariance or Correlation matrix is utilized. These matrices can be calculated using the data matrix. It involves a mathematical procedure known as Eigen analysis; normally, after normalizing (zero-mean) the data matrix for each characteristic, the analysis can be carried out via eigenvalue decomposition of a data covariance (or correlation) matrix of a data matrix. The fundamental purpose of principal component analysis (PCA) is to eliminate unwanted signal components by performing an orthogonal projection on the data[201]. We reduced the 2040 time stamps of each experiment using PCA.

SVM classifier: In BCI research, Support Vector Machine (SVM) is regarded as one of the most accurate classifiers[202][203]. SVM differentiates between classes using hyperplanes or groups of hyperplanes in a very high (or even infinite) dimensional space. Performance of a particular linear SVM is based on the trade-off parameter C, which balances the relative value of minimizing training error and maximizing margins between classes, which directly affect the classifier's generalizability. The kernel determines the classification accuracy of an SVM-based classifier. BCI systems often

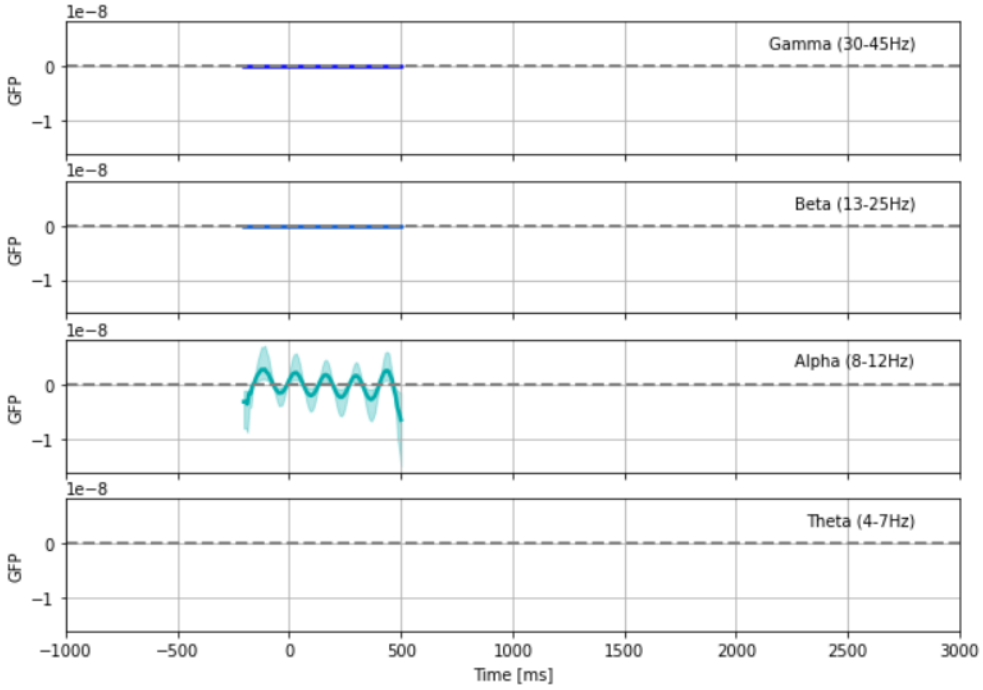


Figure 4.11: Wave bands visualization.

employ a Gaussian kernel or a Radial Base Function (RBF)[204]. Cross-validation improves the accuracy of the model by preventing overfitting. In K-fold cross validation, data are initially separated into k segments or folds of equal (or nearly equal) size. Cross-validation is used in this instance to determine the ideal RBF kernel parameters C and gamma and to estimate the model's performance. In this experiment, a multiclass approach with 10-fold cross-validation was implemented in the RBF kernel SVM system. With C equal to 400 and gamma equal to $4 * 10_{-5}$, the best results were obtained.

Results: Six individuals were tested, three of whom were healthy and three of whom were drug dependent. The precision and recall obtained during the testing phase after completing each of the four tasks are displayed in Table 1. Six individuals were tested, three of whom were healthy and three of whom were drug dependent. High precision and recall suggest that the model has a high level of performance.

These outcomes indicated outstanding performance, especially for the last three patients (healthy patients). Nevertheless, the first three patients executed the four tasks with less precision than the healthy subjects. High precision and recall suggest that the

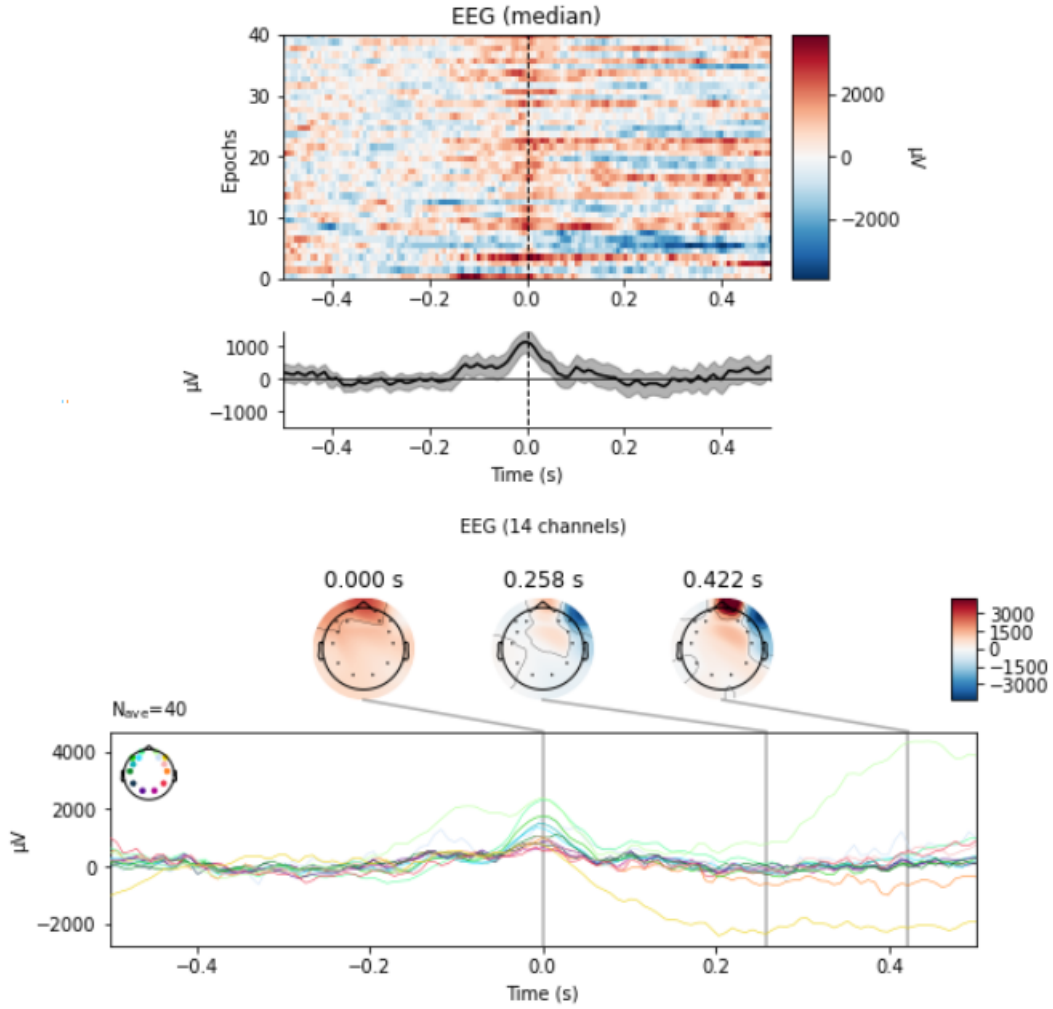


Figure 4.12: Visualization of FC5 data after preprocessing.

model has a high level of performance.

4.5.2 Attention-based Bi-LSTM

Bidirectional Long Short-Term Memory: LSTM has established its presence in the field of sequence signal analysis by sharing weights[205]. It is capable of learning long-term dependencies and adjusting for the issue of disappearing gradients successfully. Unlike the conventional unidirectional LSTM, the bidirectional LSTM is capable of capturing dynamic information from both earlier and later EEG sequence segments [206]. The bidirectional LSTM network consists of a forward layer and a backward layer, as shown in Figure 1. *ht* is the hidden output sequence of the forward layer, which is computed using EEG samples from time index 1 to t. LSTM stores informa-

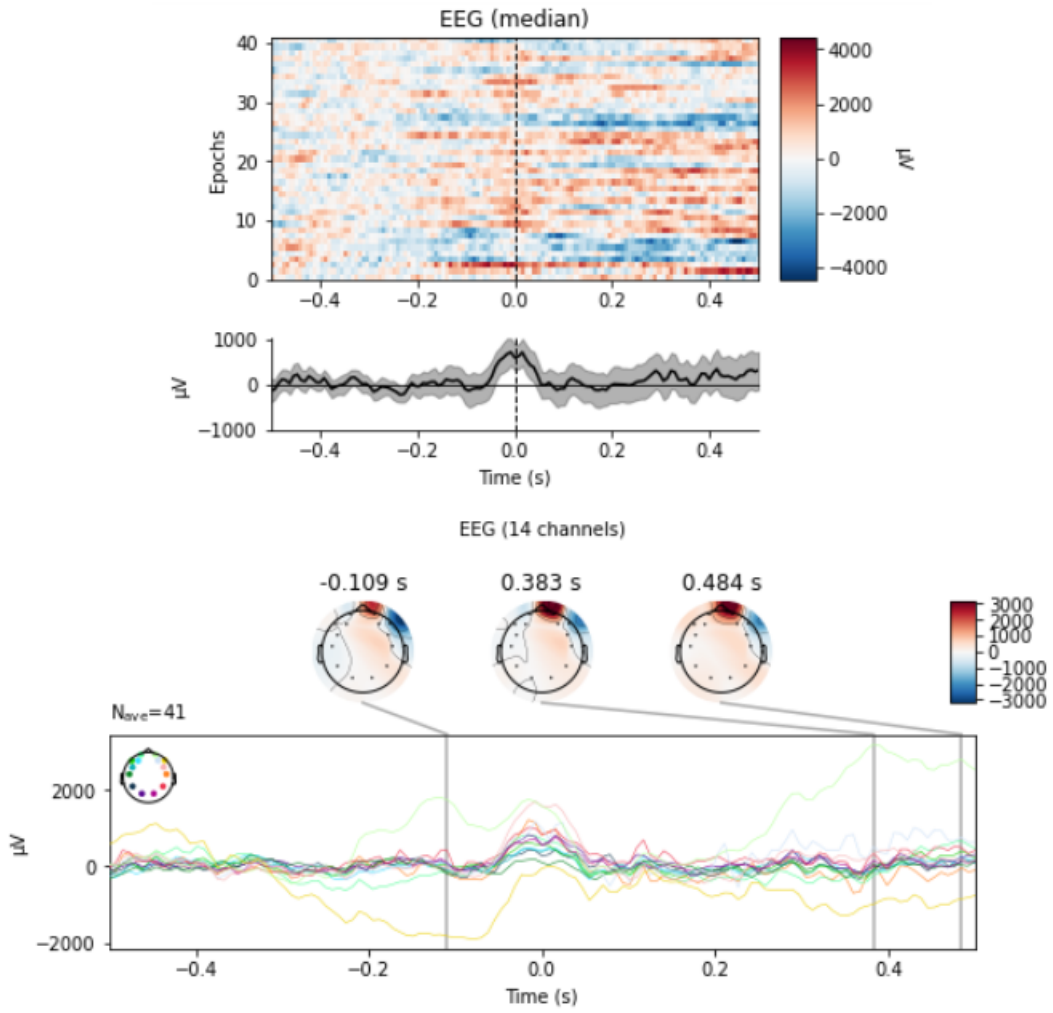


Figure 4.13: Visualization of FC6 data after preprocessing.

tion using three custom-built gates[207]. In Hochreiter's[205] original architecture, the update to the cell output state depends on the previous output of the hidden layer and the current input. In addition, they employed the previous cell condition as a parameter and attached a peephole connection. The data flow between the gates and inputs of a single LSTM cell is seen in Figure 2. At each time t, the current input is xt, the prior hidden state is $ht-1$, and the previous output state of the cell is $ct-1$. These equations can be used to determine the outputs of three gates:

$$i_t = \delta(X_i x_t + H_i h_t - 1 + C_i c_t - 1 + b_i) \tag{4.1}$$

$$o_t = \delta(X_0 x_t + H_0 h_t - 1 + C_0 c_t - 1 + b_0) \tag{4.2}$$

```
Filtering raw data in 1 contiguous segment
Setting up band-pass filter from 0.5 - 50 Hz

FIR filter parameters
---------------------
Designing a one-pass, zero-phase, non-causal bandpass filter:
- Windowed time-domain design (firwin) method
- Hamming window with 0.0194 passband ripple and 53 dB stopband attenuation
- Lower passband edge: 0.50
- Lower transition bandwidth: 0.50 Hz (-6 dB cutoff frequency: 0.25 Hz)
- Upper passband edge: 50.00 Hz
- Upper transition bandwidth: 12.50 Hz (-6 dB cutoff frequency: 56.25 Hz)
- Filter length: 845 samples (6.602 sec)

Effective window size : 16.000 (s)
```

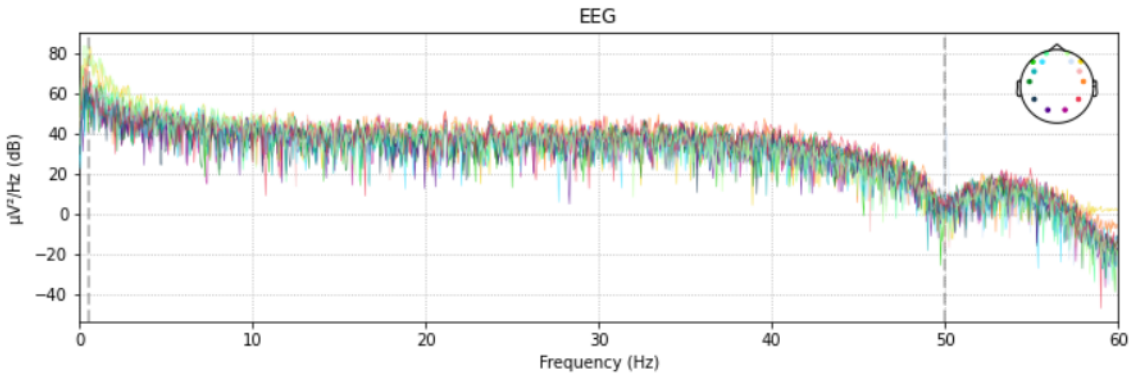


Figure 4.14: Data Visualization Before cleaning (red) and after cleaning (black).

$$f_t = \delta(X_f x_t + H_f h_t - 1 + C_f c_t - 1 + b_f) \tag{4.3}$$

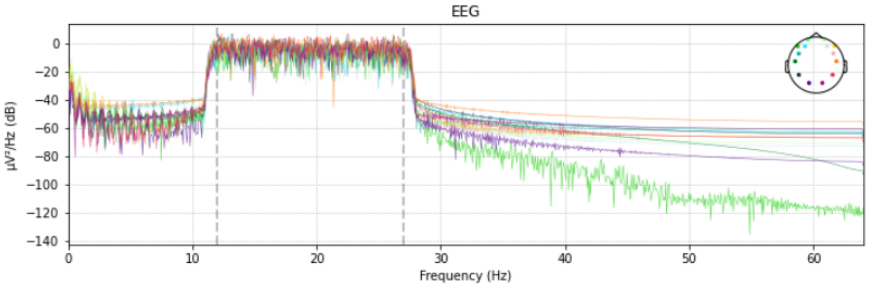
The forget gate ft determines if the $ct-1$ was retained, the input gate it determines if the state was modified by the current input xt, and the output gate ot determines if the $ht-1$ was transmitted to the next cell. At each timestamp t, at is the contender for updating the memory cell. Using the following equations, calculate the output of the current LSTM cell ct and the current hidden state ht.

$$a_t = \delta(X_a x_t + H_a h_t - 1 + C_a c_t - 1 + b_a) \tag{4.4}$$

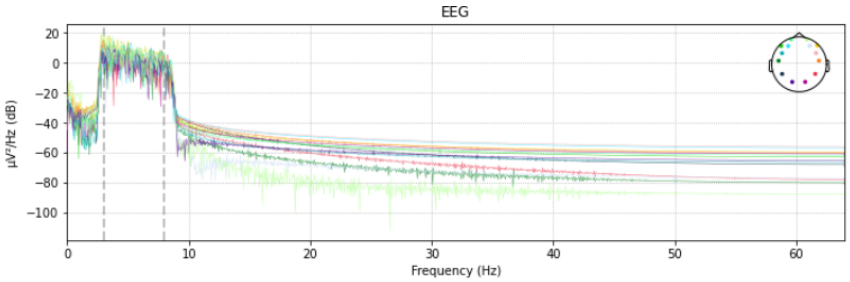
$$c_t = f_t \bigotimes c_t - 1 + i_i \bigotimes a_t \tag{4.5}$$

$$h_t = o_t \bigotimes tanh(c_t) \tag{4.6}$$

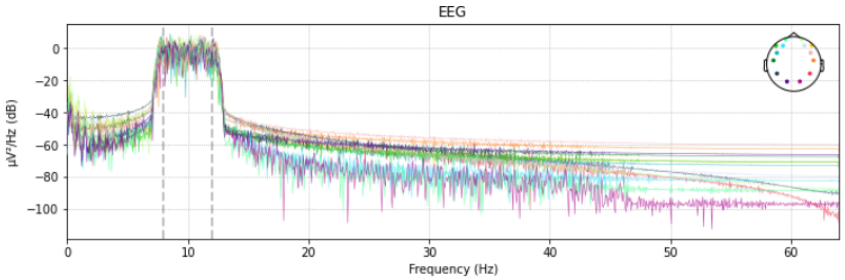
Attention Mechanism: Imitated from the human visual system, the Attention mechanism is vital to Computer Vision (CV), Natural Language Processing (NLP), and Au-



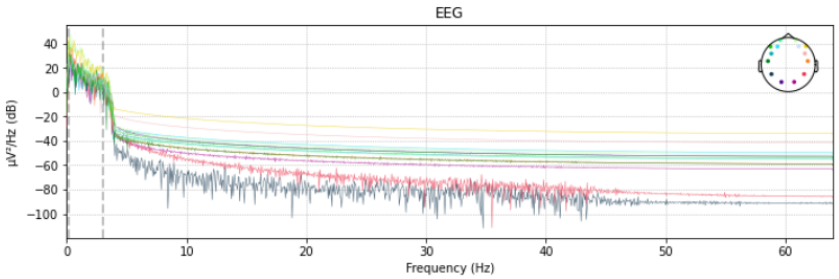
(a)



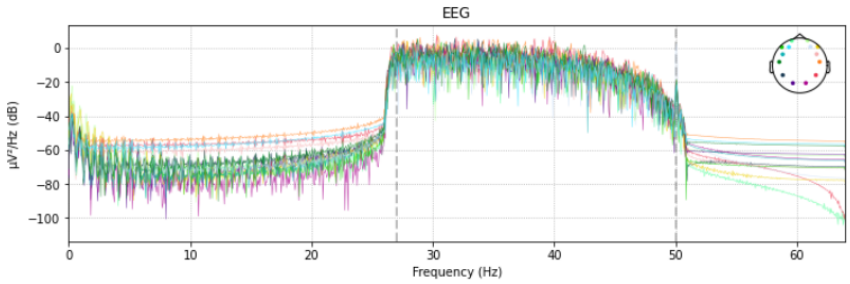
(b)



(c)



(d)

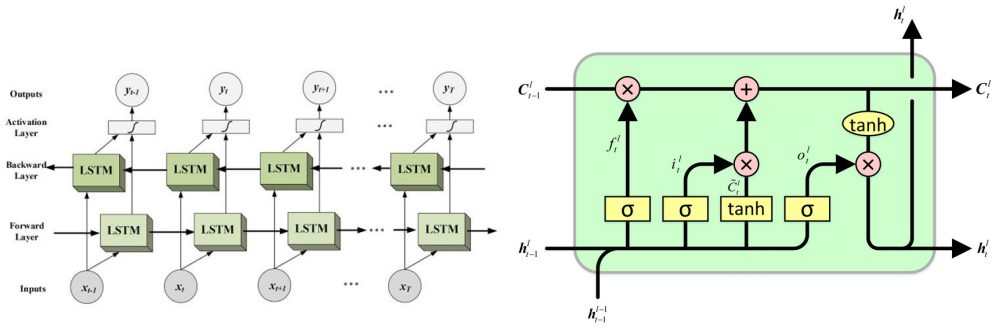


(e)

Figure 4.15: (a) Final Beta Wave (b) Final Theta Wave (c) Final Alpha Wave (d) Final Delta Wave (e) Final Gamma Wave

Table 4.1: The precision, recall of the four classes using SVM.

		LH	LL	RH	RL
Patient 01	Precision	0.712	0.721	0.719	0.720
	Recall	0.717	0.712	0.725	0.713
Patient 02	Precision	0.715	0.721	0.709	0.710
	Recall	0.722	0.723	0.714	0.705
Patient 03	Precision	0.720	0.716	0.702	0.706
	Recall	0.721	0.719	0.709	0.712
Patient 04	Precision	0.782	0.784	0.779	0.776
	Recall	0.790	0.788	0.775	0.781
Patient 05	Precision	0.775	0.779	0.770	0.772
	Recall	0.781	0.782	0.769	0.776
Patient 06	Precision	0.781	0.782	0.776	0.762
	Recall	0.775	0.786	0.773	0.769



(a) (b)

Figure 4.16: (a) Illustration of Bi-LSTM network (b) Illustration of LSTM Cell

tomatic Speech Recognition (ASR) [208][209][210][211]. Not all signals contribute to classification in the same way. Thus, an Attention mechanism s(t) is jointly trained as a weighted sum of the BiLSTM, with the Attention output dependent on the weights.

$$u_{(t)} = tanh(W_w y_{(t)} + b_w) \tag{4.7}$$

$$\alpha_{(t)} = \frac{exp({u_{(t)}}^T u_w)}{\sum exp({u_{(t)}}^T u_w)} \tag{4.8}$$

$$s_{(t)} = \sum \alpha_{(t)} y_{(t)} \tag{4.9}$$

Following the Fully-connected layer $u_{(t)}$ for learning features of $y_{(t)}$ is a Softmax layer $\alpha_{(t)}$ that provides a probability distribution. The values W_W, u_w, and b_w reflect

trainable weights and bias, accordingly. It selects and extracts the most relevant temporal and spatial information from $y_{(t)}$ by multiplying $\alpha_{(t)}$ according to its contribution to decoding tasks. A major advantage of the attention mechanism is that it provides our model with a deep mining feature extraction.

Experimental setup The preprocessing and feature extraction were identical to those described previously, with one exception: the filters were applied to all 14 channels, as opposed to just C3 and C4 in the previous section, and the data now contains 14 channels instead of 2. The proposed model consists of one improved Bi-LSTM layer, an attention weighting layer, two completely linked layers, and a softmax classification layer. At each timestamp t, all channels were utilized concurrently as input and sent to the Bi-LSTM layer, followed by the application of the attention method to its output. On layers with complete connectivity, the dropout approach is used to prevent overfitting, and the activation function is ReLU. Each patient was trained on four tasks, therefore the output size of the final, fully-linked layer is set to 4. Finally, the largest output value index is deemed the robot's direction decision. The hyperparameters (such as the dropout rate and regularization coefficient) were found via trial and error. The network parameters were updated using stochastic gradient descent and the Adam optimizer.

Table 4.2: Hyper-parameters of our proposed model

No	Hyper-parameter	Value
1	Bi-LSTM Layers	1
2	Attention Layers	1
3	Fully connected Layers	2
4	Train Set	80%
5	Test Set	20%
6	Optimizer	Adam
7	Learning Rate	10^{-3}
8	Dropout	0.2
9	Batch Size	32

Results Experiments were conducted using an AMD Ryzen 7 4800H 2.90 GHz computer with 16GB of RAM and a Gtx 1660Ti GPU in this investigation. The code was

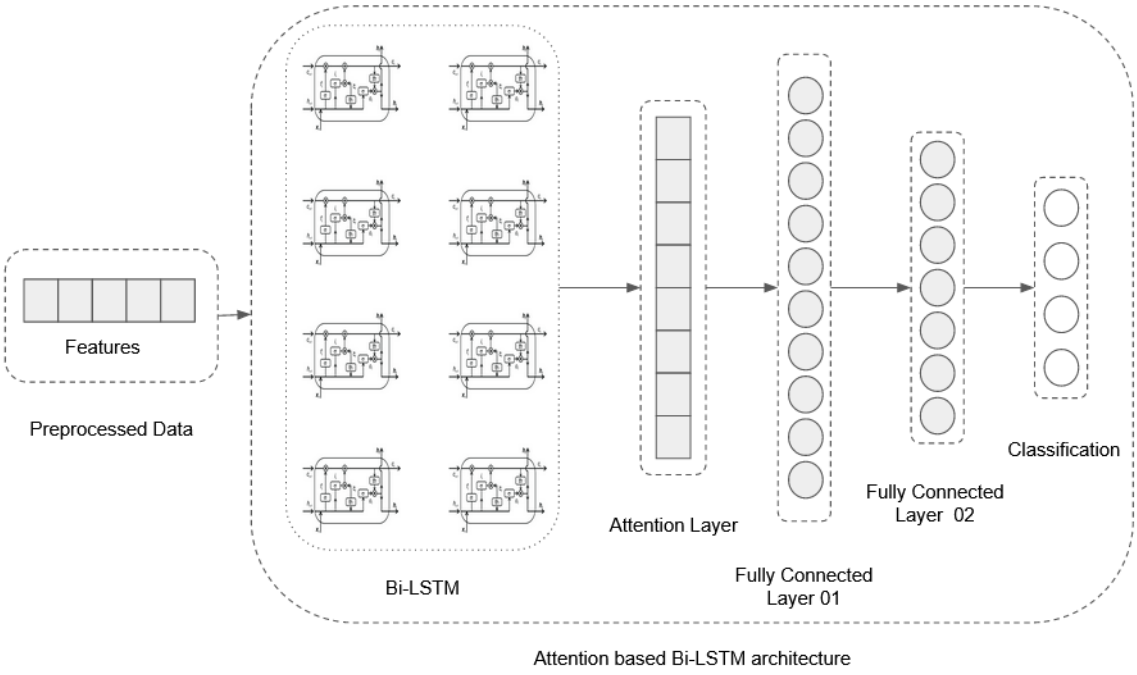


Figure 4.17: Attention based Bi-LSTM architecture

written in Keras with a backend of Tensorflow. After training the model with 200 epochs, the model is finally tested using test data to offer a valid assessment of the model's precision and recall, which at the last epoch have reached a saturation level of approximately 81%. Table 2 provides a summary of the accuracy, recall, and overall average obtained during the testing phase after completing each of the four tasks of our methodology. Six individuals were tested, three of whom were healthy and three of whom were drug dependent. Examining the confusion matrix linked with Figure 6 will provide additional information regarding this classification. These outcomes demonstrated exceptional performance. However, the first three patients performed the four tasks with less accuracy than the healthy participants. High precision and recall are indicative of the model's superior performance. According to the confusion matrix, the average accuracy of the testing phase is 81.60 percent, and the model is completely confused between the linked tasks (left and right hand, left and right legs) rather than the other tasks.

Table 4.3: The precision, recall of the four classes and the total average using attention based Bi-LSTM.

		LH	LL	RH	RL
Patient 01	Precision	0.776	0.744	0.769	0.794
	Recall	0.772	0.782	0.790	0.813
Patient 02	Precision	0.751	0.801	0.762	0.771
	Recall	0.774	0.752	0.786	0.792
Patient 03	Precision	0.764	0.795	0.754	0.798
	Recall	0.798	0.804	0.782	0.803
Patient 04	Precision	0.836	0.852	0.842	0.846
	Recall	0.813	0.846	0.844	0.841
Patient 05	Precision	0.851	0.857	0.849	0.850
	Recall	0.856	0.854	0.852	0.843
Patient 06	Precision	0.864	0.859	0.829	0.830
	Recall	0.842	0.841	0.832	0.842
Average	Precision	0.807	0.818	0.800	0.814
	Recall	0.809	0.813	0.814	0.822

4.6 Graph Convolutional Neural Network

4.6.1 Mathematical Background

The majority of current models that decode EEG using Deep Learning, notably CNNs (e.g. EEG-Net), employ the Euclidean coordinates of a given EEG electrode in a manner analogous to how picture coordinates are used by a conventional CNN. Nevertheless, a graphical representation of intra-cortical connectivity (particularly correlation) may be a more meaningful input for characterizing the topological relationship (connectivity) between network nodes [1]. Alternative connection measures to correlation (such as coherence) have not yet been investigated for this Graph Convolutional Network (GCN) idea. In the meantime, Layerwise Relevance Propagation (LRP) has been recently applied to EEG classification problems [2], providing visual insight on inner network mechanics, which can explain classification errors and facilitate important modifications, as well as providing a more detailed analysis of neural mechanisms.

Spectral Convolution: As described in [3], the ideal convolution in the case of GCN is spectral rather than spatial. This form of convolutional filter has the advantage of

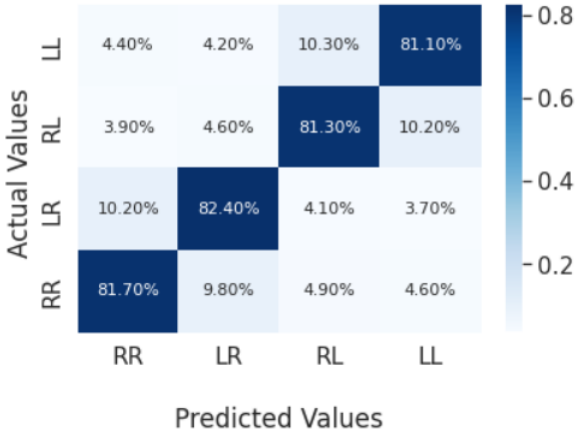


Figure 4.18: Confusion matrix obtained from testing

identifying local characteristics that are independent of Euclidean coordinates, and GCNs [4] typically prefer this.

Node Localization: Therefore, the connection matrix (adjacency matrix), W_{ij} can provide more information regarding the structure of nodes than Euclidean coordinates. W_{ij} is derived using Spearman's power correlation between nodes in our scenario (electrodes). The diagonal degree matrix is$D_{ii} = W_{ij}$, and the Laplacian L is $L = D - W$. The Laplacian is diagonalized by the Fourier basis U so that $L = UUT$ is a diagonal matrix of eigenvalues. The Fourier basis U is known as the graph Fourier transform. In conclusion, the pairwise node connectivity is related to the Laplacian, and the Laplacian is related to U, which is utilized to transform the input into the spectral convolution domain.

Implementing Convolution: In the Fourier domain, the graph convolution (denoted by $*_G$) of signals x and y is defined as:

$$h_t = o_t \bigotimes tanh(c_t) \tag{4.10}$$

To assure localisation of the convolution filter and enhance its computational efficiency, a polynomial filter:

$$g_\theta(\Lambda) = \sum_{k=0}^{K-1} \theta_k \Lambda^k \tag{4.11}$$

A vector of polynomial coefficients θ is formed. The Chebyshev family is an obvious option, as its coefficients can be learned recursively and it is frequently used to create filters for signal processing. In summary, each convolutional layer consists of filters whose coefficients represent the parameters that can be learned. (i) RGB values $\rightarrow$ node spectral features, (ii) spatial localization $\rightarrow$ connectivity between nodes, and (iii) learned NN weights $\rightarrow$ Chebyshev polynomial coefficients. Concerning the final point, the backpropagation algorithm computes the (in our case, cross-entropy) loss in relation to the filter coefficients. In addition, all the standard network parameter options (pooling, activation function, optimization, learning rate, etc.) are applicable to GCNs, therefore the same code structure as CNNs can be utilized. In order to reduce dimensionality, pooling, also known as graph coarsening in this context, causes the signal properties of well-connected nodes to be summed, similar to how localized picture information is added in a CNN.

4.7 Diverse Features Graph Convolutional Neural Network

4.7.1 Implementation

Processing: Each raw recording was processed as follows: a common subset of bipolar montage electrodes was picked from the raw channels, the recording was re-sampled to 250Hz, then a highpass filter at 1Hz, and a notch filter at the power-line frequency of 50Hz was applied. We note that neither typical physiological EEG artifacts such as eye blinks or muscle movements nor defective channels were eliminated directly. Implementation was accomplished using MNE-python library methods. We separated the preprocessed recordings into 10s windows that did not overlap. As previously established, each window consists of an EEG recording from 14 bipolar channels. We adopt the simplification assumption that each 1s window's signal is indepen-

dent of other windows in the same recording. Notably, while ML training is conducted using windows, window predictions are combined to generate subject predictions.

Feature Extraction: In this model, each trial is regarded to be an independent graph with 14 electrodes as nodes and bidirectional fully linked links between each electrode pair as edges; therefore, categorization of motor imagery has been performed in a graph classification manner. Before feature extraction, the raw EEG data were pre-processed using StandardScaler normalization based solely on the mean and standard deviation of the training data (see train-valid-test segmentation details in the model details subsection under Model 2). To compute the trial's feature matrix, we extracted temporal, spectral, and connectivity properties from the trial's normalized EEG signals for each node (channel). Mean Absolute Value (MAV), Variance, Mean Square Root (MSR), Root Mean Square (RMS), Log Detector (LD), Waveform Length (WL), Difference Absolute Standard Deviation Value (DASDV), Zero Crossing (ZC), Skewness, and Kurtosis were computed for the temporal features according to [6]. For the frequency characteristics, the mean power density of the five conventional frequency bands Delta (0.5-4 Hz), Theta (4-8 Hz), Alpha (8-12 Hz), Beta (12-35 Hz), and Gamma (¿35 Hz) was computed. For the connectivity characteristics, the mean and standard deviation of MS-coherence over each electrode pair (182 pairings in total, excluding self-coherence) were derived from the same five frequency bands. Therefore, the size of the feature matrix for each trial is 14x25, which includes 10 time features, 5 frequency features, and 10 connection features for each node (channel). The characteristics were normalized with StandardScaler before to being input to the GCN model, as ZC has a substantially larger scale than other features.

Architecture: Although conventional deep learning techniques such as Long short-term memory (LSTM) and Convolutional Neural Network (CNN) have been extensively implemented in the similar field of research (EEG-based motor imagery classification), these models disregard the connectivity and strength of the connectivity between electrode pairs. In this study, we propose a Graphical Convolutional Network

(GCN) that not only takes connectivity and its strength into account, but also uses the spatial feature extraction power of CNN in order to decode the underlying neurophysiological properties from motor imaging tasks.

Our proposed GCN model is comprised of three Chebyshev spectral graph convolutional operators, each containing a Chebyshev convolutional (ChebConv), a Batch Normalization, and a Rectified Linear Unit (ReLU) layer. On each ChebConv layer, the GCN has 64, 128, and 264 filters with 3, 4, and 5 filter sizes, respectively. In the conclusion, we forward feed the concatenated outputs of global average pooling and global maximum pooling to a dense layer with 512 cells, followed by a Log Softmax activation, to achieve the class prediction probability (task).

Table 4.4: Hyper-parameters of GCN Model

No	Hyper-parameter	Value
1	Number of Epochs	200
2	Window Size	0.5
3	Train Set	70%
4	Test Set	30%
5	Optimizer	Adam
6	Learning Rate	$4 * 10^{-4}$
7	Dropout	0.2
8	Batch size	64
9	Loss Function	Cross-Entropy
10	Regularization	10^{-3}

Table 4.5: Architecture of the proposed GCN Model

Layer	Type	Parameters
CC1	Chebyshev convolutional (ChebConv)	Filters : 64, K = 3
BL1	Batch Normalization1d	Features: 64 Momentum = 0.1
CC2	Chebyshev convolutional (ChebConv)	Filters : 128, K = 4
BL2	Batch Normalization1d	Features: 128 Momentum = 0.1
CC3	Chebyshev convolutional (ChebConv)	Filters : 256, K = 5
BL3	Batch Normalization1d	Features: 256 Momentum = 0.1
RL	ReLU Layer	-
DN	Dense Layer	512
AC	Log SoftMax Activation Function	-

```
GCN(
  (conv1): ChebConv(100, 64, K=3, normalization=sym)
  (conv2): ChebConv(64, 128, K=4, normalization=sym)
  (conv3): ChebConv(128, 256, K=5, normalization=sym)
  (bn1): BatchNorm1d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (bn2): BatchNorm1d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (bn3): BatchNorm1d(256, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (dense): Linear(in_features=512, out_features=4, bias=True)
)
```

Figure 4.19: Screenshot of the proposed architecture

```
Starting training...
Saving..
Saving..
Saving..


epoch#: 10 | train loss: 0.845 | val loss: 0.976 | train acc: 43.441 | val acc: 52.644
Best Acc: 61.67 | Best Epoch: 10 | epochs_no_improve: 0
Device = cuda:0; Time per iter: 1.052 seconds
Saving..
Saving..
```

(a)

```
epoch#: 200| train loss: 0.092 | val loss: 0.989 | train acc: 86.210 | val acc: 85.805
Best Acc: 86.999 | Best Epoch: 126| epochs_no_improve: 26
Device = cuda:0; Time per iter: 1.142 seconds
Early stopping!
```

(b)

Figure 4.20: (a) Results after 10 epochs (b) Results after 200 epochs

4.7.2 Results

The suggested GCN model with different features achieves 87 percent validation accuracy, 86.43 percent precision, 86.01 percent recall, and 86.21 percent F1-score. Due to the minimal number of trainable parameters, the training method is so efficient that training per iteration takes 1,1 seconds. At epoch 80, the suggested model converges (convergence time: 117s). The details of training and performance are displayed in the accuracy plot. The accuracy plot shows that the training and validation accuracy converge with the same rate to the same values, thus the model fits well.

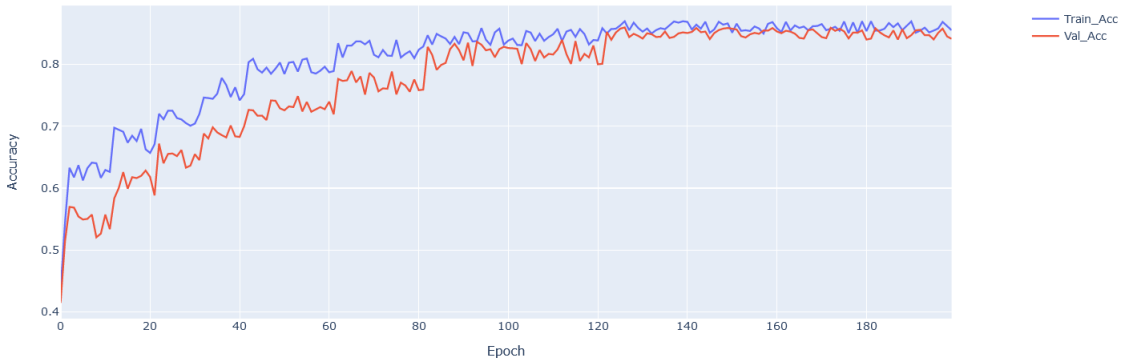


Figure 4.21: Train and validation accuracy.

Table 4.6: Classification Report

Class	Precision	Recall	F1-Score
Left Hand	0.8742	0.8709	0.8725
Right Hand	0.8739	0.8697	0.8718
Left Leg	0.8523	0.8486	0.8504
Right Leg	0.8569	0.8512	0.8540

4.8 Time Domain Graph Convolutional Neural Netowk

4.8.1 Preprocessing

Each raw recording was processed as follows: a common subset of bipolar montage electrodes was picked from the raw channels, the recording was resampled to 250Hz, then a highpass filter at 1Hz, and a notch filter at the power-line frequency of 50Hz was applied. We note that neither typical physiological EEG artifacts such as eye blinks or muscle movements nor defective channels were eliminated directly. Implementation was accomplished using MNE-python library methods. We separated the preprocessed recordings into 0.5s windows that did not overlap. As previously established, each window consists of an EEG recording from 14 bipolar channels. We adopt the simplification assumption that each 1s window's signal is independent of other windows in the same recording. Notably, while ML training is conducted using windows, window predictions are combined to generate subject predictions.

4.8.2 Feature Extraction

The Power Spectral Density (PSD) was used to summarize the frequency content of the win- dowed EEG signals into the primary brain wave bands: delta (1-4Hz), theta (4-7.5Hz), alpha (7.5-13Hz), lower beta (13- 16Hz), higher beta (16-30Hz), and gamma (30-40Hz). We extracted the total band power from each of the 8 modulation channels, resulting in a feature matrix of shape (14 channels 6 features) for each window.

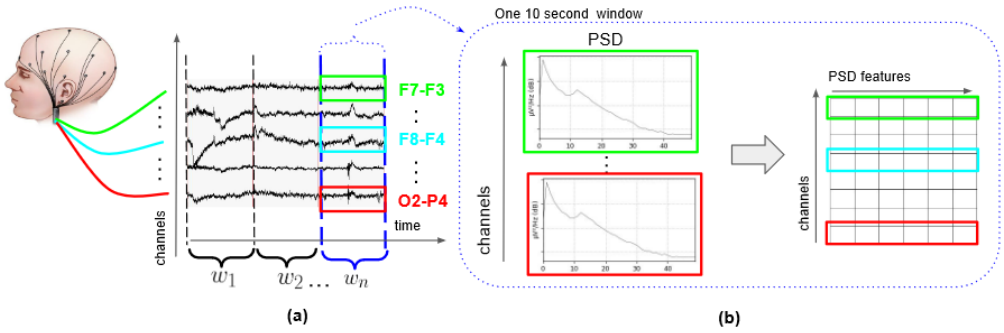


Figure 4.22: Feature Extraction technique inspired from[16].

4.8.3 Architecture

Our goal is to train a TimeDomain-GCN to classify 4 classes: left hand, right hand, left reg, right leg. The preceding assertion holds valid for both hypothetical and actual facts. Separate consideration was given to simulated and actual data in 4-class classification jobs.

The initial input dimensions are $N_{subjects} * N_{classes} * N_{segments} * N_{features} * N_{channels} = 17 * 4 * 32 * 6 * 14$. However, this is reconfigured so that the first four dimensions are merged, resulting in dimensions equal to $17 * 4 * 32 * 6 * 14 = 13056 * 14$. Therefore, all time samples for all participants, all classes, all seconds, and all timestamps have been merged into a single dimension, N. Then, a N minus 1 element vector containing related labels is generated. Finally, the data features and labels are randomized, resulting in N random time samples with 14 channel values each. This can be depicted using N instantaneous brain signal heat maps. The electrodes FC5 and FC6 show a

high activity (motor imagery area). We also find that, particularly in those electrodes, hypothetical patterns are scaled-down counterparts of implemented patterns. Before feeding the input to the model, the Spearman Power correlation is computed for all concatenated time samples (not shuffled, in Nx14 format), followed by the calculation of the Adjacency and Laplacian Matrices. The Laplacian is then used to execute the graph Fourier transform on the GCN input, while the adjacency matrix is used to specify how coarsening operations (similar to pooling in a ConvNet) are applied at each layer. Figure 2 displays the Adjacency and Laplacian Matrices produced from the Spearman correlation.

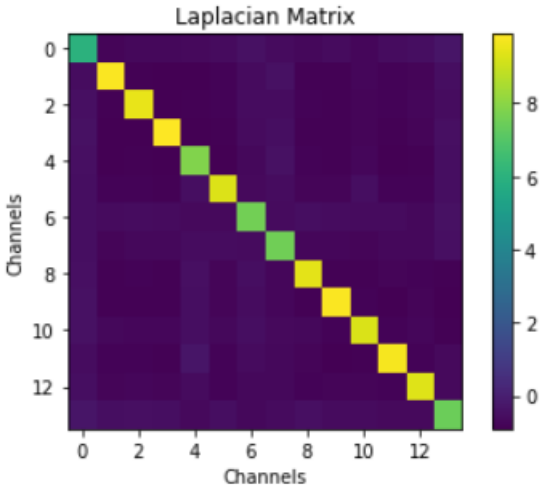


Figure 4.23: Laplacian matrix of a subject.

4.8.4 Training

All the experiments above were performed and executed on the previously mentioned laptop.

4.8.5 Results

The model is evaluated on six individuals, three of whom are healthy and three of whom are not. The suggested GCN model with varied characteristics achieves 91% validation accuracy, 91.16% precision, 90.75% recall, and 90.95% F1-score. Due to

Table 4.7: Architecture of the proposed second GCN Model

Layer	Type	Parameters	Activation
CC1	Chebyshev convolutional (ChebConv)	Filters : 16	SoftPlus
PL1	Graph Pooling Layer	MAX	-
BL1	Batch Normalization1d	Features: 16 Momentum = 0.1	-
CC2	Chebyshev convolutional (ChebConv)	Filters : 32	SoftPlus
PL2	Graph Pooling Layer	MAX	-
BL2	Batch Normalization1d	Features: 32 Momentum = 0.1	-
CC3	Chebyshev convolutional (ChebConv)	Filters : 64	SoftPlus
PL3	Graph Pooling Layer	MAX	-
BL3	Batch Normalization1d	Features: 64 Momentum = 0.1	-
CC4	Chebyshev convolutional (ChebConv)	Filters : 128	SoftPlus
PL4	Graph Pooling Layer	MAX	-
BL4	Batch Normalization1d	Features: 128 Momentum = 0.1	-
CC5	Chebyshev convolutional (ChebConv)	Filters : 256	SoftPlus
PL5	Graph Pooling Layer	MAX	-
SM	Softmax Layer	-	Softmax

Table 4.8: Hyper-parameters of second GCN Model

No	Hyper-parameter	Value
1	Number of Epochs	200
2	Window Size	0.5
3	Train Set	85%
4	Test Set	15%
5	Optimizer	Adam
6	Learning Rate	10^{-7}
7	Batch size	16
8	Loss Function	Cross-Entropy
9	L2 Norm	10^{-7}

the small number of trainable parameters, the training method is so effective that training per iteration requires only 1.4 seconds. The recommended model converges at epoch 120. The accuracy plot displays training and performance-related information. The accuracy plot demonstrates that the training and validation accuracy converge to the same values at the same rate, indicating that the model is well-fitting. From the confusion matrix we notice that the model reduced the confusion between the inner classes (Legs and Hands), and almost killed the confusion between outer classes.

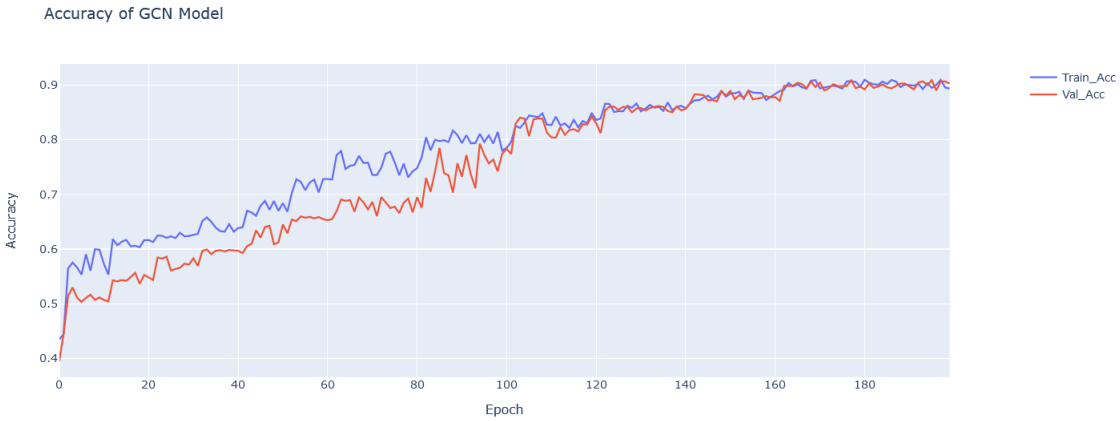


Figure 4.24: Train and validation accuracy.

Table 4.9: Model Evaluation

Subject	Precision	Recall	F1-Score
1	93.85%	93.42%	93.63%
2	94.31%	93.95%	94.13%
3	89.47%	89.63%	89.55%
4	88.81%	88.22%	88.51%
5	92.62%	91.54%	92.08%
6	87.92%	87.79%	87.85%
Average	91.16%	90.75%	90.95%

4.9 Discussion

4.9.1 Evaluation study:

Regarding the dataset of drug-dependent individuals, which may significantly impact the model's precision. Our suggested method consists of two models, the first of which is based on machine learning (SVM) and the second on deep learning (Attention-based Bi-LSTM), and achieves high performance and classification accuracy. Due to the fact that the noisy data (drug effects) are not geometrically separable, the first SVM-based model has only been applied to two filtered channels (C3 and C4) and has obtained roughly 74% precision. In contrast, the second model that applies attention-based Bi-LSTM to the 14 channels demonstrates greater precision than the first model, which is one of the advantages of our design (it provides deep mining). In this classification task, the evaluation study indicates that the second model is more accurate than the

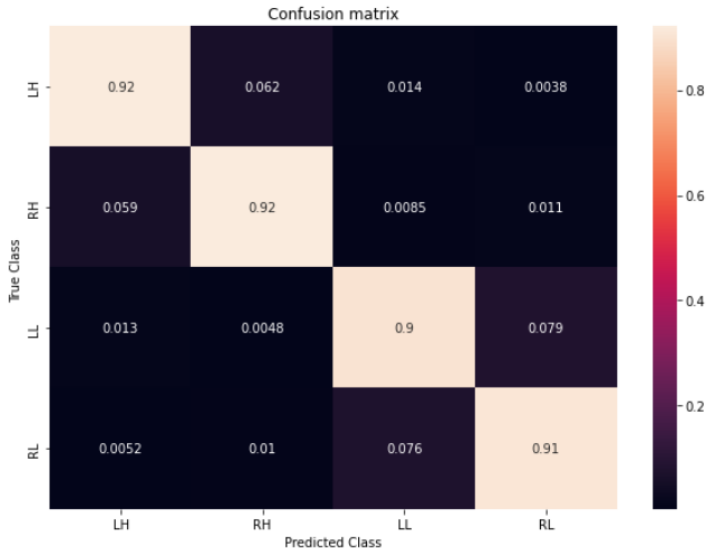


Figure 4.25: The Confusion Matrix.

first. We propose that the attention mechanism allowed Bi-LSTM achieve these outcomes compared to SVM, taking into account the abnormalities caused by the drug-induced changes in the values of beta and gamma (generally). In an experiment involving the classification of EEG data, LSTM designs fared better than cutting-edge methods. There are two reasons behind the results. First, EEG signals often comprise components that are different and difficult to distinguish. Because they are statistical features, the spatial-frequency features accurately reflect ”categorizable portions” that are simple to identify. However, because ”hard sections” are not linear and do not remain constant, statistical characteristics cannot be used to model them successfully[212][213]. We observe that our results are better after applying the first GCN model with diverse features and observing the similarity convergence of both train and validation accuracy which means that the model is fitting well. The major advantage of using GCN is to detect the non-linearity in our patterns and provide a deep mining technique that helped us to increase the precision, recall, and F1-score to approximately 86%. This average illustrates that the classification for unhealthy subjects reduces the previous measures. The final Time Domain GCN model provided is capable of meeting the challenge of subject-specific adaptation. It obtained competitive

results, with an average of 91 % for accuracy, precision, recall, and F1-score. The above positive results revealed that the introduced technique successfully filtered raw EEG signals and classified MI tasks including the abnormal signals.

4.9.2 Comparative study:

As noted, the relative precisions obtained from references[214][215] were 82.70 percent and 88.00 percent. Our model exceeds theirs in terms of classification accuracy and number of classes. Article [216] utilized an additional dataset of healthy subjects, and the experiment utilized all 128 channels. The achieved categorization performance was 84.00 percent. Our model is able to maintain and extract the prior long-term and short-term sequence characteristics, as well as account for the dynamic linkage between the past and future to the present in electrical activation. Our algorithm extracts the most significant EEG segment from a full EEG recording. The subject-level prediction comparison between the presented GCN methods and competitive models[217][218][219][91][220][221]] was proposed. The presented method has enhanced classification precision and achieved state-of-the-art results. The exceptional performance was a result of the feature extraction method's ability to extract useful features from raw EEG signals. The following GCN model identified features successfully by collaborating with the topological link between overall features. The last model shows a high classification for all the subjects, thus we conclude that we successfully achieved the goal of this study of studying different medical backgrounds and generalizing the brain-controlling technique. Those results will be used later to control robots in 4 directions by mapping the classes with robot degrees of freedom.

4.10 Contributions

The first contribution obtained from our study is to collect an important EEG dataset that contains several medical backgrounds. Second, we have achieved the first results ever that classified motor imagery for both normal and abnormal EEG signals of healthy and unhealthy subjects. The third contribution is that we evaluated our model

so that we achieved very respectable results. The fourth contribution is to build a powerful multi-method approach which is able to classify motor imagery for both healthy and unhealthy subjects in real-time, thus achieving another goal of our study. Finally, my biggest contribution is to work with some of the best professors and scientists in Algeria and Italy and publish some scientific papers together.

Chapter 5

Conclusion and future works

This thesis reports a novel multi-method approach that uses simultaneous models such as SVM, attention-based Bi-LSTM, and GCN to classify motor imagery tasks for healthy and unhealthy (drugged, alcoholic) patients. Our test results showed that the proposed approach achieved significant improvement in classifying human decisions to control robots over some other approaches that have been applied to only healthy patients. It shows that our model with an attention mechanism did a good job of catching and separating EEG features that were different because of how drugs affected the brain, but the most accurate models were the gcn, which achieved very respectable results. However, we succeeded in speeding up models, thus controlling robots in real time.

In future research, we will use other revolutionary methods, such as combining EOG and EEG, to control multiple robots. We will also study the relationship between the precision of those models on all the channels and the models that learn only from specific channels (like the example of SVM and channels C3 and C4). Finally, we will apply a benchmark with several machine learning and deep learning models applied to our dataset with several preprocessing and feature extraction methods. The further objective will be studying the effects of Parkinson's on EEG signals and trying to improve their lives by facilitating robots and smart houses to control them.

REFERENCES

[1] X. Zhang, L. Yao, X. Wang, J. Monaghan, D. Mcalpine, and Y. Zhang, "A survey on deep learning-based non-invasive brain signals: recent advances and new frontiers," *Journal of neural engineering*, vol. 18, no. 3, p. 031002, 2021.

[2] J. Henshaw, "Improved brain-computer interface methods with application to gaming," Ph.D. dissertation, University of Sheffield, 2017.

[3] H. Lee and S. Choi, "Pca+ hmm+ svm for eeg pattern classification," in *Seventh International Symposium on Signal Processing and Its Applications, 2003. Proceedings.*, vol. 1. IEEE, 2003, pp. 541–544.

[4] L. George, F. Lotte, R. V. Abad, and A. Lécuyer, "Using scalp electrical biosignals to control an object by concentration and relaxation tasks: design and evaluation," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2011, pp. 6299–6302.

[5] J. Mercier, "Contribution to the study of the use of brain-computer interfaces in virtual and augmented reality," Ph.D. dissertation, INSA de Rennes, 2015.

[6] H. Piitulainen, A. Botter, R. Merletti, and J. Avela, "Multi-channel electromyography during maximal isometric and dynamic contractions," *Journal of Electromyography and Kinesiology*, vol. 23, no. 2, pp. 302–310, 2013.

[7] A. Bulling, D. Roggen, and G. Tröster, "Eyemote–towards context-aware gaming using eye movements recorded from wearable electrooculography," in *International Conference on Fun and Games*. Springer, 2008, pp. 33–45.

[8] G. Pfurtscheller, T. Solis-Escalante, R. Ortner, P. Linortner, and G. R. Muller-Putz, "Self-paced operation of an ssvep-based orthosis with and without an imagery-based "brain switch:" a feasibility study towards a hybrid bci," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 18, no. 4, pp. 409–414, 2010.

[9] M. Xu, H. Qi, B. Wan, T. Yin, Z. Liu, and D. Ming, "A hybrid bci speller paradigm combining p300 potential and the ssvep blocking feature," *Journal of neural engineering*, vol. 10, no. 2, p. 026001, 2013.

[10] X. Yong, M. Fatourechi, R. K. Ward, and G. E. Birch, "The design of a point-and-click system by integrating a self-paced brain–computer interface with an eye-tracker," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 1, no. 4, pp. 590–602, 2011.

[11] J. Rantanen, N. Alfthan, J. Impio, T. Karinsalo, M. Malmivaara, R. Matala, M. Makinen, A. Reho, P. Talvenmaa, M. Tasanen *et al.*, "Smart clothing for the arctic environment," in *digest of papers. fourth international symposium on wearable computers*. IEEE, 2000, pp. 15–23.

[12] C.-C. Postelnicu and D. Talaba, "P300-based brain-neuronal computer interaction for spelling applications," *IEEE transactions on biomedical engineering*, vol. 60, no. 2, pp. 534–543, 2012.

[13] J. LaRocco, M. D. Le, and D.-G. Paeng, "A systemic review of available low-cost eeg headsets used for drowsiness detection," *Frontiers in neuroinformatics*, p. 42, 2020.

[14] A. Craik, Y. He, and J. L. Contreras-Vidal, "Deep learning for electroencephalogram (eeg) classification tasks: a review," *Journal of neural engineering*, vol. 16, no. 3, p. 031001, 2019.

[15] Q. Lin, S.-q. Ye, X.-m. Huang, S.-y. Li, M.-z. Zhang, Y. Xue, and W.-S. Chen, "Classification of epileptic eeg signals with stacked sparse autoencoder based on deep learning," in *International conference on intelligent computing*. Springer, 2016, pp. 802–810.

[16] N. Wagh and Y. Varatharajah, "Eeg-gcnn: Augmenting electroencephalogram-based neurological disease diagnosis using a domain-guided graph convolutional neural network," in *Machine Learning for Health*. PMLR, 2020, pp. 367–378.

[17] F. Putze, A. Vourvopoulos, A. Lécuyer, D. Krusienski, S. Bermúdez i Badia, T. Mullen, and C. Herff, "Editorial: Brain-computer interfaces and augmented/virtual reality," *Frontiers in Human Neuroscience*, vol. 14, 2020. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnhum.2020.00144

[18] J. Wolpaw, N. Birbaumer, W. Heetderks, D. McFarland, P. Peckham, G. Schalk, E. Donchin, L. Quatrano, C. Robinson, and T. Vaughan, "Brain-computer interface technology: a review of the first international meeting," *IEEE Transactions on Rehabilitation Engineering*, vol. 8, no. 2, pp. 164–173, 2000.

[19] R. P. Rao and R. Scherer, "Brain-computer interfacing [in the spotlight]," *IEEE Signal Processing Magazine*, vol. 27, no. 4, pp. 152–150, 2010.

[20] V. Mihajlović, B. Grundlehner, R. Vullers, and J. Penders, "Wearable, wireless eeg solutions in daily life applications: What are we missing?" *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 1, pp. 6–21, 2015.

[21] M. Bamdad, H. Zarshenas, and M. A. Auais, "Application of bci systems in neurorehabilitation: a scoping review," *Disability and Rehabilitation:*

Assistive Technology, vol. 10, no. 5, pp. 355–364, 2015. [Online]. Available: https://doi.org/10.3109/17483107.2014.961569

[22] L. F. Nicolas-Alonso and J. Gomez-Gil, "Brain computer interfaces, a review," *Sensors*, vol. 12, no. 2, pp. 1211–1279, 2012. [Online]. Available: https://www.mdpi.com/1424-8220/12/2/1211

[23] W. Ma, D. Tran, T. Le, H. Lin, and S.-M. Zhou, "Using eeg artifacts for bci applications," in *2014 International Joint Conference on Neural Networks (IJCNN)*, 2014, pp. 3628–3635.

[24] M. A. Lebedev and M. A. Nicolelis, "Brain–machine interfaces: past, present and future," *Trends in Neurosciences*, vol. 29, no. 9, pp. 536–546, 2006. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0166223606001470

[25] N. Naseer and K.-S. Hong, "Corrigendum "fnirs-based brain-computer interfaces: a review"," *Frontiers in Human Neuroscience*, vol. 9, 2015. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnhum.2015.00172

[26] P. Suppes, Z.-L. Lu, and B. Han, "Brain wave recognition of words," *Proceedings of the National Academy of Sciences*, vol. 94, no. 26, pp. 14 965–14 969, 1997. [Online]. Available: https://www.pnas.org/doi/abs/10.1073/pnas.94.26.14965

[27] "Niedermeyer, e. and da silva, f. l. (2005). electroencephalography: basic principles, clinical applications, and related fields. lippincott williams wilkins."

[28] J. Henshaw, "Improved brain-computer interface methods with application to gaming," 2017.

[29] D. Steyrl, R. J. Kobler, G. R. Müller-Putz *et al.*, "On similarities and differences of invasive and non-invasive electrical brain signals in brain-computer interfacing," *Journal of biomedical science and engineering*, vol. 9, no. 08, p. 393, 2016.

[30] H. Gray and W. H. Lewis, *Anatomy of the human body*. Philadelphia, Lea Febiger, 1918, https://www.biodiversitylibrary.org/bibliography/20311. [Online]. Available: https://www.biodiversitylibrary.org/item/60234

[31] D. J. McFarland and J. R. Wolpaw, "Brain-computer interfaces for communication and control," *Commun. ACM*, vol. 54, no. 5, p. 60–66, may 2011. [Online]. Available: https://doi.org/10.1145/1941487.1941506

[32] H. Lee and S. Choi, "Pca+hmm+svm for eeg pattern classification," in *Seventh International Symposium on Signal Processing and Its Applications, 2003. Proceedings.*, vol. 1, 2003, pp. 541–544 vol.1.

[33] S. Bunce, M. Izzetoglu, K. Izzetoglu, B. Onaral, and K. Pourrezaei, "Functional near-infrared spectroscopy," *IEEE Engineering in Medicine and Biology Magazine*, vol. 25, no. 4, pp. 54–62, 2006.

[34] M. Hämäläinen, R. Hari, R. J. Ilmoniemi, J. Knuutila, and O. V. Lounasmaa, "Magnetoencephalography—theory, instrumentation, and applications to noninvasive studies of the working human brain," *Rev. Mod. Phys.*, vol. 65, pp. 413–497, Apr 1993. [Online]. Available: https://link.aps.org/doi/10.1103/RevModPhys.65.413

[35] S. Ogawa, D. W. Tank, R. Menon, J. M. Ellermann, S. G. Kim, H. Merkle, and K. Ugurbil, "Intrinsic signal changes accompanying sensory stimulation: functional brain mapping with magnetic resonance imaging." *Proceedings of the National Academy of Sciences*, vol. 89, no. 13, pp. 5951–5955, 1992. [Online]. Available: https://www.pnas.org/doi/abs/10.1073/pnas.89.13.5951

[36] E. Leuthardt, K. Miller, G. Schalk, R. Rao, and J. Ojemann, "Electrocorticography-based brain computer interface-the seattle experience," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 2, pp. 194–198, 2006.

[37] E. Niedermeyer, "The electrocerebellogram," *Clinical EEG and Neuroscience*, vol. 35, no. 2, pp. 112–115, 2004, pMID: 15164822. [Online]. Available: https://doi.org/10.1177/155005940403500213

[38] Y. Futagi, T. Ishihara, K. Tsuda, Y. Suzuki, and M. Goto, "Theta rhythms associated with sucking, crying, gazing and handling in infants," *Electroencephalography and clinical neurophysiology*, vol. 106, no. 5, pp. 392–399, 1998.

[39] J. R. Wolpaw and D. J. McFarland, "Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans," *Proceedings of the national academy of sciences*, vol. 101, no. 51, pp. 17 849–17 854, 2004.

[40] W. T. Blume, "Drug effects on eeg," *Journal of Clinical Neurophysiology*, vol. 23, no. 4, pp. 306–311, 2006.

[41] P. Aricò, G. Borghini, G. Di Flumeri, N. Sciaraffa, and F. Babiloni, "Passive bci beyond the lab: current trends and future directions," *Physiological measurement*, vol. 39, no. 8, p. 08TR02, 2018.

[42] G. Pfurtscheller and C. Neuper, "Motor imagery and direct brain-computer communication," *Proceedings of the IEEE*, vol. 89, no. 7, pp. 1123–1134, 2001.

[43] D. Huang, K. Qian, D.-Y. Fei, W. Jia, X. Chen, and O. Bai, "Electroencephalography (eeg)-based brain–computer interface (bci): A 2-d virtual wheelchair control based on event-related desynchronization/synchronization and state con-

trol," *IEEE transactions on Neural Systems and Rehabilitation engineering*, vol. 20, no. 3, pp. 379–388, 2012.

[44] H. Cecotti and A. J. Ries, "Best practice for single-trial detection of event-related potentials: Application to brain-computer interfaces," *International Journal of Psychophysiology*, vol. 111, pp. 156–169, 2017.

[45] D. Regan, "Steady-state evoked potentials," *JOSA*, vol. 67, no. 11, pp. 1475–1489, 1977.

[46] G. Pfurtscheller and F. L. Da Silva, "Event-related eeg/meg synchronization and desynchronization: basic principles," *Clinical neurophysiology*, vol. 110, no. 11, pp. 1842–1857, 1999.

[47] C. Guger, G. Edlinger, and G. Krausz, "Hardware/software components and applications of bcis," *recent advances in brain-computer interface systems*, pp. 1–24, 2011.

[48] E. Donchin, K. M. Spencer, and R. Wijesinghe, "The mental prosthesis: assessing the speed of a p300-based brain-computer interface," *IEEE transactions on rehabilitation engineering*, vol. 8, no. 2, pp. 174–179, 2000.

[49] M. Inami, N. Kawakami, D. Sekiguchi, Y. Yanagida, T. Maeda, and S. Tachi, "Visuo-haptic display using head-mounted projector," in *Proceedings IEEE Virtual Reality 2000 (Cat. No. 00CB37048)*. IEEE, 2000, pp. 233–240.

[50] U. Hoffmann, J.-M. Vesin, T. Ebrahimi, and K. Diserens, "An efficient p300-based brain–computer interface for disabled subjects," *Journal of Neuroscience methods*, vol. 167, no. 1, pp. 115–125, 2008.

[51] S. Sutton, M. Braren, J. Zubin, and E. John, "Evoked-potential correlates of stimulus uncertainty," *Science*, vol. 150, no. 3700, pp. 1187–1188, 1965.

[52] N. V. Manyakov, N. Chumerin, A. Combaz, and M. M. Van Hulle, "Comparison of classification methods for p300 brain-computer interface on disabled subjects," *Computational intelligence and neuroscience*, vol. 2011, 2011.

[53] M. Fabiani, G. Gratton, D. Karis, E. Donchin *et al.*, "Definition, identification, and reliability of measurement of the p300 component of the event-related brain potential," *Advances in psychophysiology*, vol. 2, no. S1, p. 78, 1987.

[54] C. Guger, S. Daban, E. Sellers, C. Holzner, G. Krausz, R. Carabalona, F. Gramatica, and G. Edlinger, "How many people are able to control a p300-based brain–computer interface (bci)?" *Neuroscience letters*, vol. 462, no. 1, pp. 94–98, 2009.

[55] M. Spüler, M. Bensch, S. Kleih, W. Rosenstiel, M. Bogdan, and A. Kübler, "Online use of error-related potentials in healthy users and people with severe

motor impairment increases performance of a p300-bci," *Clinical Neurophysiology*, vol. 123, no. 7, pp. 1328–1337, 2012.

[56] C. S. Herrmann, "Human eeg responses to 1–100 hz flicker: resonance phenomena in visual cortex and their potential correlation to cognitive phenomena," *Experimental brain research*, vol. 137, no. 3, pp. 346–353, 2001.

[57] H. Bakardjian, T. Tanaka, and A. Cichocki, "Optimization of ssvep brain responses with application to eight-command brain–computer interface," *Neuroscience letters*, vol. 469, no. 1, pp. 34–38, 2010.

[58] G. R. Burkitt, R. B. Silberstein, P. J. Cadusch, and A. W. Wood, "Steady-state visual evoked potentials and travelling waves," *Clinical Neurophysiology*, vol. 111, no. 2, pp. 246–258, 2000.

[59] S. G. Thorpe, P. L. Nunez, and R. Srinivasan, "Identification of wave-like spatial structure in the ssvep: Comparison of simultaneous eeg and meg," *Statistics in medicine*, vol. 26, no. 21, pp. 3911–3926, 2007.

[60] J. Ding, G. Sperling, and R. Srinivasan, "Attentional modulation of ssvep power depends on the network tagged by the flicker frequency," *Cerebral cortex*, vol. 16, no. 7, pp. 1016–1029, 2006.

[61] A. J. McDaid, S. Xing, and S. Q. Xie, "Brain controlled robotic exoskeleton for neurorehabilitation," in *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. IEEE, 2013, pp. 1039–1044.

[62] T. Sakurada, T. Kawase, K. Takano, T. Komatsu, and K. Kansaku, "A bmi-based occupational therapy assist suit: asynchronous control by ssvep," *Frontiers in neuroscience*, vol. 7, p. 172, 2013.

[63] G. Vanacker, J. d. R. Millán, E. Lew, P. W. Ferrez, F. G. Moles, J. Philips, H. Van Brussel, and M. Nuttin, "Context-based filtering for assisted brain-actuated wheelchair driving," *Computational intelligence and neuroscience*, vol. 2007, 2007.

[64] C. J. Bell, P. Shenoy, R. Chalodhorn, and R. P. Rao, "Control of a humanoid robot by a noninvasive brain–computer interface in humans," *Journal of neural engineering*, vol. 5, no. 2, p. 214, 2008.

[65] R. S. Fisher, G. Harding, G. Erba, G. L. Barkley, and A. Wilkins, "Photic- and pattern-induced seizures: a review for the epilepsy foundation of america working group," *Epilepsia*, vol. 46, no. 9, pp. 1426–1441, 2005.

[66] D. Makri, C. Farmaki, and V. Sakkalis, "Visual fatigue effects on steady state visual evoked potential-based brain computer interfaces," in *2015 7th Interna-*

tional IEEE/EMBS Conference on Neural Engineering (NER). IEEE, 2015, pp. 70–73.

[67] C. Mahl, G. Hayrettin, P. Danny, E. Marieke, S. Lasse, D. Matthieu, and A. Alexandra, "Bacteriahunt: Evaluating multi-paradigm bci interaction," *Journal on Multimodal User Interfaces*, vol. 1, no. 4, pp. 11–25, 2010.

[68] S. I. Hjelm and C. Browall, "Brainball-using brain activity for cool competition," in *Proceedings of NordiCHI*, vol. 7, no. 9, 2000, p. 19.

[69] B. Hamadicharef, H. Zhang, C. Guan, C. Wang, K. S. Phua, K. P. Tee, and K. K. Ang, "Learning eeg-based spectral-spatial patterns for attention level measurement," in *2009 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2009, pp. 1465–1468.

[70] B. Z. Allison, J. Jin, Y. Zhang, and X. Wang, "A four-choice hybrid p300/ssvep bci for improved accuracy," *Brain-Computer Interfaces*, vol. 1, no. 1, pp. 17–26, 2014.

[71] P. Horki, T. Solis-Escalante, C. Neuper, and G. Müller-Putz, "Combined motor imagery and ssvep based bci control of a 2 dof artificial upper limb," *Medical & biological engineering & computing*, vol. 49, no. 5, pp. 567–577, 2011.

[72] J. Long, Y. Li, T. Yu, and Z. Gu, "Target selection with hybrid feature for bci-based 2-d cursor control," *IEEE Transactions on biomedical engineering*, vol. 59, no. 1, pp. 132–140, 2011.

[73] E. Yin, Z. Zhou, J. Jiang, F. Chen, Y. Liu, and D. Hu, "A speedy hybrid bci spelling approach combining p300 and ssvep," *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 2, pp. 473–483, 2013.

[74] L. George and A. Lécuyer, "An overview of research on" passive" brain-computer interfaces for implicit human-computer interaction," in *International Conference on Applied Bionics and Biomechanics ICABB 2010-Workshop W1" Brain-Computer Interfacing and Virtual Reality"*, 2010.

[75] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain–computer interfaces for communication and control," *Clinical neurophysiology*, vol. 113, no. 6, pp. 767–791, 2002.

[76] G. Pfurtscheller, B. Z. Allison, G. Bauernfeind, C. Brunner, T. Solis Escalante, R. Scherer, T. O. Zander, G. Mueller-Putz, C. Neuper, and N. Birbaumer, "The hybrid bci," *Frontiers in neuroscience*, vol. 4, p. 3, 2010.

[77] Y. Li, J. Long, T. Yu, Z. Yu, C. Wang, H. Zhang, and C. Guan, "An eeg-based bci system for 2-d cursor control by combining mu/beta rhythm and p300 potential,"

IEEE Transactions on Biomedical Engineering, vol. 57, no. 10, pp. 2495–2505, 2010.

[78] L. R. Young and D. Sheena, "Survey of eye movement recording methods," *Behavior research methods & instrumentation*, vol. 7, no. 5, pp. 397–429, 1975.

[79] W. S. Wijesoma, K. S. Wee, O. C. Wee, A. P. Balasuriya, K. T. San, and K. K. Soon, "Eog based control of mobile assistive platforms for the severely disabled," in *2005 IEEE International Conference on Robotics and Biomimetics-ROBIO*. IEEE, 2005, pp. 490–494.

[80] L. H. Opie, "Cellular basis for therapeutic choices in heart failure," pp. 2559–2561, 2004.

[81] R. Leeb, M. Lancelle, V. Kaiser, D. W. Fellner, and G. Pfurtscheller, "Thinking penguin: multimodal brain–computer interface control of a vr game," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 5, no. 2, pp. 117–128, 2013.

[82] H. Riechmann, N. Hachmeister, H. Ritter, and A. Finke, "Asynchronous, parallel on-line classification of p300 and erd for an efficient hybrid bci," in *2011 5th International IEEE/EMBS Conference on Neural Engineering*. IEEE, 2011, pp. 412–415.

[83] R. Leeb, H. Sagha, R. Chavarriaga, and J. del R Millán, "A hybrid brain–computer interface based on the fusion of electroencephalographic and electromyographic activities," *Journal of neural engineering*, vol. 8, no. 2, p. 025011, 2011.

[84] C. Davatzikos, K. Ruparel, Y. Fan, D. Shen, M. Acharyya, J. W. Loughead, R. C. Gur, and D. D. Langleben, "Classifying spatial patterns of brain activity with machine learning methods: application to lie detection," *Neuroimage*, vol. 28, no. 3, pp. 663–668, 2005.

[85] K. Choi and A. Cichocki, "Control of a wheelchair by motor imagery in real time," in *International conference on intelligent data engineering and automated learning*. Springer, 2008, pp. 330–337.

[86] K. LaFleur, K. Cassady, A. Doud, K. Shades, E. Rogin, and B. He, "Quadcopter control in three-dimensional space using a noninvasive motor imagery-based brain–computer interface," *Journal of neural engineering*, vol. 10, no. 4, p. 046003, 2013.

[87] R. Bousseta, I. El Ouakouak, M. Gharbi, and F. Regragui, "Eeg based brain computer interface for controlling a robot arm movement through thought," *Irbm*, vol. 39, no. 2, pp. 129–135, 2018.

[88] Y. Chae, J. Jeong, and S. Jo, "Toward brain-actuated humanoid robots: asynchronous direct control using an eeg-based bci," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1131–1144, 2012.

[89] B. E. Olivas-Padilla and M. I. Chacon-Murguia, "Classification of multiple motor imagery using deep convolutional neural networks and spatial filters," *Applied Soft Computing*, vol. 75, pp. 461–472, 2019.

[90] P. Wang, A. Jiang, X. Liu, J. Shang, and L. Zhang, "Lstm-based eeg classification in motor imagery tasks," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 26, no. 11, pp. 2086–2095, 2018.

[91] R. Zhang, Q. Zong, L. Dou, and X. Zhao, "A novel hybrid deep learning scheme for four-class motor imagery classification," *Journal of neural engineering*, vol. 16, no. 6, p. 066004, 2019.

[92] Y. Hou, S. Jia, X. Lun, Y. Shi, and Y. Li, "Deep feature mining via attention-based bilstm-gcn for human motor imagery recognition," *arXiv preprint arXiv:2005.00777*, 2020.

[93] J. L. Collinger, B. Wodlinger, J. E. Downey, W. Wang, E. C. Tyler-Kabara, D. J. Weber, A. J. McMorland, M. Velliste, M. L. Boninger, and A. B. Schwartz, "High-performance neuroprosthetic control by an individual with tetraplegia," *The Lancet*, vol. 381, no. 9866, pp. 557–564, 2013.

[94] L. R. Hochberg, D. Bacher, B. Jarosiewicz, N. Y. Masse, J. D. Simeral, J. Vogel, S. Haddadin, J. Liu, S. S. Cash, P. Van Der Smagt *et al.*, "Reach and grasp by people with tetraplegia using a neurally controlled robotic arm," *Nature*, vol. 485, no. 7398, pp. 372–375, 2012.

[95] V. S. Polikov, P. A. Tresco, and W. M. Reichert, "Response of brain tissue to chronically implanted neural electrodes," *Journal of neuroscience methods*, vol. 148, no. 1, pp. 1–18, 2005.

[96] S. Finnigan, R. G. O'Connell, T. D. Cummins, M. Broughton, and I. H. Robertson, "Erp measures indicate both attention and working memory encoding decrements in aging," *Psychophysiology*, vol. 48, no. 5, pp. 601–611, 2011.

[97] K. E. Mathewson, T. J. Harrison, and S. A. Kizuk, "High and dry? comparing active dry eeg electrodes to active and passive wet electrodes," *Psychophysiology*, vol. 54, no. 1, pp. 74–82, 2017.

[98] M. Hämäläinen, R. Hari, R. J. Ilmoniemi, J. Knuutila, and O. V. Lounasmaa, "Magnetoencephalography—theory, instrumentation, and applications to noninvasive studies of the working human brain," *Reviews of modern Physics*, vol. 65, no. 2, p. 413, 1993.

[99] J. Soul and A. du Plessis, "New technologies in pediatric neurology," *Semin Pediatr Neurol*, vol. 6, pp. 101–110, 1999.

[100] B.-K. Min, M. J. Marzelli, and S.-S. Yoo, "Neuroimaging-based approaches in the brain–computer interface," *Trends in biotechnology*, vol. 28, no. 11, pp. 552–560, 2010.

[101] D. A. Boas, A. M. Dale, and M. A. Franceschini, "Diffuse optical imaging of brain activation: approaches to optimizing image sensitivity, resolution, and accuracy," *Neuroimage*, vol. 23, pp. S275–S288, 2004.

[102] C.-S. Wei, Y.-T. Wang, C.-T. Lin, and T.-P. Jung, "Toward drowsiness detection using non-hair-bearing eeg-based brain-computer interfaces," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 26, no. 2, pp. 400–406, 2018.

[103] G. Li and W.-Y. Chung, "Estimation of eye closure degree using eeg sensors and its application in driver drowsiness detection," *Sensors*, vol. 14, no. 9, pp. 17 491–17 515, 2014.

[104] C.-T. Lin, C.-H. Chuang, C.-S. Huang, S.-F. Tsai, S.-W. Lu, Y.-H. Chen, and L.-W. Ko, "Wireless and wearable eeg system for evaluating driver vigilance," *IEEE Transactions on biomedical circuits and systems*, vol. 8, no. 2, pp. 165–176, 2014.

[105] B. Farnsworth, "Vr: Training and performance [part 2]."

[106] O. E. Krigolson, C. C. Williams, A. Norton, C. D. Hassall, and F. L. Colino, "Choosing muse: Validation of a low-cost, portable eeg system for erp research," *Frontiers in neuroscience*, vol. 11, p. 109, 2017.

[107] M. Doudou, A. Bouabdallah, and V. Cherfaoui, "A light on physiological sensors for efficient driver drowsiness detection system," *Sensors & Transducers Journal*, vol. 224, no. 8, pp. 39–50, 2018.

[108] F. Mohamed, S. K. Nataraj, S. F. Ahmed, and S. Yaacob, "An approach in determining fatigueness and drowsiness detection using eeg," *Res. Inventy Int. J. Eng. Sci*, vol. 8, no. 3, pp. 20–28, 2018.

[109] J. Murphy and C. Russomanno, "Openbci: Biosensing for everybody," *Kickstarter. Available online at: www. kickstarter. com/projects/openbci/openbci-biosensing-for-everybody*, 2016.

[110] P. de Lissa, S. Sörensen, N. Badcock, J. Thie, and G. McArthur, "Measuring the face-sensitive n170 with a gaming eeg system: a validation study," *Journal of neuroscience methods*, vol. 253, pp. 47–54, 2015.

[111] N. A. Badcock, P. Mousikou, Y. Mahajan, P. De Lissa, J. Thie, and G. McArthur, "Validation of the emotiv epoc® eeg gaming system for measuring research quality auditory erps," *PeerJ*, vol. 1, p. e38, 2013.

[112] N. A. Badcock, K. A. Preece, B. de Wit, K. Glenn, N. Fieder, J. Thie, and G. McArthur, "Validation of the emotiv epoc eeg system for research quality auditory event-related potentials in children," *PeerJ*, vol. 3, p. e907, 2015.

[113] A. Manolova, G. Tsenov, V. Lazarova, and N. Neshov, "Combined eeg and emg fatigue measurement framework with application to hybrid brain-computer interface," in *2016 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*. IEEE, 2016, pp. 1–5.

[114] S. Sanei and J. Chambers, "Introduction to eeg," *EEG signal processing*, pp. 1–34, 2007.

[115] R. P. Rao, *Brain-computer interfacing: an introduction*. Cambridge University Press, 2013.

[116] H.-T. Hsu, I.-H. Lee, H.-T. Tsai, H.-C. Chang, K.-K. Shyu, C.-C. Hsu, H.-H. Chang, T.-K. Yeh, C.-Y. Chang, and P.-L. Lee, "Evaluate the feasibility of using frontal ssvep to implement an ssvep-based bci in young, elderly and als groups," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 5, pp. 603–615, 2015.

[117] S. M. T. Müller, T. F. Bastos-Filho, and M. Sarcinelli-Filho, "Monopolar and bipolar electrode settings for ssvep-based brain-computer interface," *Journal of Medical and Biological Engineering*, vol. 35, no. 4, pp. 482–491, 2015.

[118] L. Suarez, E. Cardozo, R. Attux, and D. Soriano, "An implementation of ssvep-bci system based on a cluster measure for feature selection," in *5th ISSNIP-IEEE Biosignals and Biorobotics Conference (2014): Biosignals and Robotics for Better and Safer Living (BRC)*. IEEE, 2014, pp. 1–6.

[119] A. Akce, J. J. Norton, and T. Bretl, "An ssvep-based brain–computer interface for text spelling with adaptive queries that maximize information gain rates," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 23, no. 5, pp. 857–866, 2014.

[120] X. Chen, Y. Wang, S. Gao, T.-P. Jung, and X. Gao, "Filter bank canonical correlation analysis for implementing a high-speed ssvep-based brain–computer interface," *Journal of neural engineering*, vol. 12, no. 4, p. 046008, 2015.

[121] N.-S. Kwak, K.-R. Müller, and S.-W. Lee, "A lower limb exoskeleton control system based on steady state visual evoked potentials," *Journal of neural engineering*, vol. 12, no. 5, p. 056009, 2015.

[122] L. Acqualagna, L. Botrel, C. Vidaurre, A. Kübler, and B. Blankertz, "Large-scale assessment of a fully automatic co-adaptive motor imagery-based brain computer interface," *PloS one*, vol. 11, no. 2, p. e0148886, 2016.

[123] C. Long, X. Lei, J. Chen, Y. Chang, A. Chen, and H. Li, "Event-related potential parameters of category and property violations during semantic category-based induction," *International Journal of Psychophysiology*, vol. 96, no. 3, pp. 141–148, 2015.

[124] H. Ramoser, J. Muller-Gerking, and G. Pfurtscheller, "Optimal spatial filtering of single trial eeg during imagined hand movement," *IEEE transactions on rehabilitation engineering*, vol. 8, no. 4, pp. 441–446, 2000.

[125] K. K. Ang, Z. Y. Chin, H. Zhang, and C. 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)*. IEEE, 2008, pp. 2390–2397.

[126] K. K. Ang, Z. Y. Chin, C. Wang, C. Guan, and H. Zhang, "Filter bank common spatial pattern algorithm on bci competition iv datasets 2a and 2b," *Frontiers in neuroscience*, vol. 6, p. 39, 2012.

[127] A. Zaitcev, G. Cook, W. Liu, M. Paley, and E. Milne, "Application of compressive sensing for eeg source localization in brain computer interfaces," in *2014 Loughborough Antennas and Propagation Conference (LAPC)*. IEEE, 2014, pp. 272–276.

[128] M. A. Klados, C. Papadelis, C. Braun, and P. D. Bamidis, "Reg-ica: a hybrid methodology combining blind source separation and regression techniques for the rejection of ocular artifacts," *Biomedical Signal Processing and Control*, vol. 6, no. 3, pp. 291–300, 2011.

[129] J. A. Urigüen and B. Garcia-Zapirain, "Eeg artifact removal—state-of-the-art and guidelines," *Journal of neural engineering*, vol. 12, no. 3, p. 031001, 2015.

[130] Z. Cashero, "Comparison of eeg preprocessing methods to improve the classification of p300 trials," *Master of Science Thesis, Colorado State University*, 2011.

[131] C. Maswanganyi, C. Tu, P. Owolawi, and S. Du, "Discrimination of motor imagery task using wavelet based eeg signal features," in *2018 International Conference on Intelligent and Innovative Computing Applications (ICONIC)*. IEEE, 2018, pp. 1–4.

[132] M. Besserve, M. Philippe, G. Florence, F. Laurent, L. Garnero, and J. Martinerie, "Prediction of performance level during a cognitive task from ongoing

eeg oscillatory activities," *Clinical neurophysiology*, vol. 119, no. 4, pp. 897–908, 2008.

[133] M. Saeidi, W. Karwowski, F. V. Farahani, K. Fiok, R. Taiar, P. Hancock, and A. Al-Juaid, "Neural decoding of eeg signals with machine learning: a systematic review," *Brain Sciences*, vol. 11, no. 11, p. 1525, 2021.

[134] G. Bin, X. Gao, Z. Yan, B. Hong, and S. Gao, "An online multi-channel ssvep-based brain–computer interface using a canonical correlation analysis method," *Journal of neural engineering*, vol. 6, no. 4, p. 046002, 2009.

[135] Z. Zhang, H. Li, and D. Mandic, "Blind source separation and artefact cancellation for single channel bioelectrical signal," in *2016 IEEE 13th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*. IEEE, 2016, pp. 177–182.

[136] C. Ieracitano, N. Mammone, A. Bramanti, S. Marino, A. Hussain, and F. C. Morabito, "A time-frequency based machine learning system for brain states classification via eeg signal processing," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–8.

[137] S. Santaji and V. Desai, "Analysis of eeg signal to classify sleep stages using machine learning," *Sleep and Vigilance*, vol. 4, no. 2, pp. 145–152, 2020.

[138] M. Sifuzzaman, M. R. Islam, and M. Ali, "Application of wavelet transform and its advantages compared to fourier transform," 2009.

[139] S. Sreeja, J. Rabha, K. Nagarjuna, D. Samanta, P. Mitra, and M. Sarma, "Motor imagery eeg signal processing and classification using machine learning approach," in *2017 International Conference on New Trends in Computing Sciences (ICTCS)*. IEEE, 2017, pp. 61–66.

[140] S. Kotte and J. K. Dabbakuti, "Methods for removal of artifacts from eeg signal: A review," in *Journal of Physics: Conference Series*, vol. 1706, no. 1. IOP Publishing, 2020, p. 012093.

[141] K. T. Sweeney, T. E. Ward, and S. F. McLoone, "Artifact removal in physiological signals—practices and possibilities," *IEEE transactions on information technology in biomedicine*, vol. 16, no. 3, pp. 488–500, 2012.

[142] R. Kher and R. Gandhi, "Adaptive filtering based artifact removal from electroencephalogram (eeg) signals," in *2016 International Conference on Communication and Signal Processing (ICCSP)*. IEEE, 2016, pp. 0561–0564.

[143] M. Murugappan, "Human emotion classification using wavelet transform and knn," in *2011 International Conference on Pattern Analysis and Intelligence Robotics*, vol. 1. IEEE, 2011, pp. 148–153.

[144] K. M. Puk, K. C. Gandy, S. Wang, and H. Park, "Pattern classification and analysis of memory processing in depression using eeg signals," in *International Conference on Brain Informatics*. Springer, 2016, pp. 124–137.

[145] X. Jiang, G.-B. Bian, and Z. Tian, "Removal of artifacts from eeg signals: a review," *Sensors*, vol. 19, no. 5, p. 987, 2019.

[146] P. K. Johal and N. Jain, "Artifact removal from eeg: A comparison of techniques," in *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*. IEEE, 2016, pp. 2088–2091.

[147] J. Castillo, S. Müller, E. Caicedo, and T. Bastos, "Feature extraction techniques based on power spectrum for a ssvep-bci," in *2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)*. IEEE, 2014, pp. 1051–1055.

[148] O. Friman, I. Volosyak, and A. Graser, "Multiple channel detection of steady-state visual evoked potentials for brain-computer interfaces," *IEEE transactions on biomedical engineering*, vol. 54, no. 4, pp. 742–750, 2007.

[149] S. N. Resalat and S. K. Setarehdan, "An improved ssvep based bci system using frequency domain feature classification," *American Journal of Biomedical Engineering*, vol. 3, no. 1, pp. 1–8, 2013.

[150] Z. Lin, C. Zhang, W. Wu, and X. Gao, "Frequency recognition based on canonical correlation analysis for ssvep-based bcis," *IEEE transactions on biomedical engineering*, vol. 53, no. 12, pp. 2610–2614, 2006.

[151] M. Nakanishi, Y. Wang, Y.-T. Wang, Y. Mitsukura, and T.-P. Jung, "Enhancing unsupervised canonical correlation analysis-based frequency detection of ssveps by incorporating background eeg," in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2014, pp. 3053–3056.

[152] Y. Zhang, G. Zhou, J. Jin, X. Wang, and A. Cichocki, "Frequency recognition in ssvep-based bci using multiset canonical correlation analysis," *International journal of neural systems*, vol. 24, no. 04, p. 1450013, 2014.

[153] S. N. Carvalho, T. B. Costa, L. F. Uribe, D. C. Soriano, G. F. Yared, L. C. Coradine, and R. Attux, "Comparative analysis of strategies for feature extraction and classification in ssvep bcis," *Biomedical Signal Processing and Control*, vol. 21, pp. 34–42, 2015.

[154] K. A. I. Aboalayon, M. Faezipour, W. S. Almuhammadi, and S. Moslehpour, "Sleep stage classification using eeg signal analysis: a comprehensive survey and new investigation," *Entropy*, vol. 18, no. 9, p. 272, 2016.

[155] S. Hengstler, S. Sand, and A. H. Costa, "Adaptive autoregressive modeling for time-frequency analysis," in *Proceedings of the Third International Conference on Information, Communications & Signal Processing (ICICS 2001)*, 2001, pp. 241–244.

[156] A. R. Mane, S. Biradar, and R. Shastri, "Review paper on feature extraction methods for eeg signal analysis," *Int. J. Emerg. Trend Eng. Basic Sci*, vol. 2, no. 1, pp. 545–552, 2015.

[157] Y. Xie and S. Oniga, "A review of processing methods and classification algorithm for eeg signal." *Carpathian Journal of Electronic & Computer Engineering*, vol. 12, no. 3, 2020.

[158] A. Shakeel, T. Tanaka, and K. Kitajo, "Time-series prediction of the oscillatory phase of eeg signals using the least mean square algorithm-based ar model," *Applied Sciences*, vol. 10, no. 10, p. 3616, 2020.

[159] A. S. Al-Fahoum and A. A. Al-Fraihat, "Methods of eeg signal features extraction using linear analysis in frequency and time-frequency domains," *International Scholarly Research Notices*, vol. 2014, 2014.

[160] A. H. Jahidin, M. M. Ali, M. N. Taib, N. M. Tahir, I. M. Yassin, and S. Lias, "Classification of intelligence quotient via brainwave sub-band power ratio features and artificial neural network," *Computer methods and programs in biomedicine*, vol. 114, no. 1, pp. 50–59, 2014.

[161] P. Singh, S. D. Joshi, R. K. Patney, and K. Saha, "The fourier decomposition method for nonlinear and non-stationary time series analysis," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 473, no. 2199, p. 20160871, 2017.

[162] K. Dragomiretskiy and D. Zosso, "Variational mode decomposition," *IEEE transactions on signal processing*, vol. 62, no. 3, pp. 531–544, 2013.

[163] C.-J. Peng, Y.-C. Chen, C.-C. Chen, S.-J. Chen, B. Cagneau, and L. Chassagne, "An eeg-based attentiveness recognition system using hilbert–huang transform and support vector machine," *Journal of Medical and Biological Engineering*, vol. 40, no. 2, pp. 230–238, 2020.

[164] E. P. Torres, E. A. Torres, M. Hernández-Álvarez, and S. G. Yoo, "Eeg-based bci emotion recognition: A survey," *Sensors*, vol. 20, no. 18, p. 5083, 2020.

[165] P. Jahankhani, V. Kodogiannis, and K. Revett, "Eeg signal classification using wavelet feature extraction and neural networks," in *IEEE John Vincent Atanasoff 2006 International Symposium on Modern Computing (JVA'06)*. IEEE, 2006, pp. 120–124.

[166] S. Aggarwal and N. Chugh, "Signal processing techniques for motor imagery brain computer interface: A review," *Array*, vol. 1, p. 100003, 2019.

[167] Y. Wang, K. C. Veluvolu, and M. Lee, "Time-frequency analysis of band-limited eeg with bmflc and kalman filter for bci applications," *Journal of neuroengineering and rehabilitation*, vol. 10, no. 1, pp. 1–16, 2013.

[168] G. Rutkowski, K. Patan, and P. Leśniak, "Comparison of time-frequency feature extraction methods for eeg signals classification," in *International Conference on Artificial Intelligence and Soft Computing*. Springer, 2013, pp. 320–329.

[169] J. Jin, R. Xiao, I. Daly, Y. Miao, X. Wang, and A. Cichocki, "Internal feature selection method of csp based on l1-norm and dempster–shafer theory," *IEEE transactions on neural networks and learning systems*, vol. 32, no. 11, pp. 4814–4825, 2020.

[170] R. Fu, M. Han, Y. Tian, and P. Shi, "Improvement motor imagery eeg classification based on sparse common spatial pattern and regularized discriminant analysis," *Journal of Neuroscience Methods*, vol. 343, p. 108833, 2020.

[171] M. R. Lakshmi, T. Prasad, and D. V. C. Prakash, "Survey on eeg signal processing methods," *International journal of advanced research in computer science and software engineering*, vol. 4, no. 1, 2014.

[172] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, and K.-R. Muller, "Optimizing spatial filters for robust eeg single-trial analysis," *IEEE Signal processing magazine*, vol. 25, no. 1, pp. 41–56, 2007.

[173] A. Bhalla and R. Agrawal, "Relevant feature extraction by combining independent components analysis and common spatial patterns for eeg based motor imagery classification," *Int. J. Eng. Res. Technol*, vol. 3, pp. 246–252, 2014.

[174] H. Cho, M. Ahn, S. Ahn, and S. C. Jun, "Invariant common spatio-spectral patterns," in *Proc. of TOBI 3rd Workshop*, 2012, pp. 31–32.

[175] G. Dornhege, B. Blankertz, M. Krauledat, F. Losch, G. Curio, and K.-R. Muller, "Combined optimization of spatial and temporal filters for improving brain-computer interfacing," *IEEE transactions on biomedical engineering*, vol. 53, no. 11, pp. 2274–2281, 2006.

[176] S. Kumar, A. Sharma, and T. Tsunoda, "An improved discriminative filter bank selection approach for motor imagery eeg signal classification using mutual information," *BMC bioinformatics*, vol. 18, no. 16, pp. 125–137, 2017.

[177] Q. Novi, C. Guan, T. H. Dat, and P. Xue, "Sub-band common spatial pattern (sbcsp) for brain-computer interface," in *2007 3rd International IEEE/EMBS Conference on Neural Engineering*. IEEE, 2007, pp. 204–207.

[178] H. Higashi and T. Tanaka, "Simultaneous design of fir filter banks and spatial patterns for eeg signal classification," *IEEE transactions on biomedical engineering*, vol. 60, no. 4, pp. 1100–1110, 2012.

[179] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*. Springer, 2006, vol. 4, no. 4.

[180] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of machine learning research*, vol. 3, no. Mar, pp. 1157–1182, 2003.

[181] G. H. John, R. Kohavi, and K. Pfleger, "Irrelevant features and the subset selection problem," in *Machine learning proceedings 1994*. Elsevier, 1994, pp. 121–129.

[182] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K.-R. Mullers, "Fisher discriminant analysis with kernels," in *Neural networks for signal processing IX: Proceedings of the 1999 IEEE signal processing society workshop (cat. no. 98th8468)*. Ieee, 1999, pp. 41–48.

[183] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their applications*, vol. 13, no. 4, pp. 18–28, 1998.

[184] T. Hofmann, B. Schölkopf, and A. J. Smola, "Kernel methods in machine learning," *The annals of statistics*, vol. 36, no. 3, pp. 1171–1220, 2008.

[185] M.-P. Hosseini, A. Hosseini, and K. Ahi, "A review on machine learning for eeg signal processing in bioengineering," *IEEE reviews in biomedical engineering*, vol. 14, pp. 204–218, 2020.

[186] S. Puntanen, "Regression analysis by example, by samprit chatterjee, ali s. hadi," *International Statistical Review*, vol. 81, no. 2, pp. 308–308, 2013.

[187] M.-P. Hosseini, M. R. Nazem-Zadeh, F. Mahmoudi, H. Ying, and H. Soltanian-Zadeh, "Support vector machine with nonlinear-kernel optimization for lateralization of epileptogenic hippocampus in mr images," in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2014, pp. 1047–1050.

[188] W. Yu, T. Liu, R. Valdez, M. Gwinn, and M. J. Khoury, "Application of support vector machine modeling for prediction of common diseases: the case of diabetes and pre-diabetes," *BMC medical informatics and decision making*, vol. 10, no. 1, pp. 1–7, 2010.

[189] I. Rish *et al.*, "An empirical study of the naive bayes classifier," in *IJCAI 2001 workshop on empirical methods in artificial intelligence*, vol. 3, no. 22, 2001, pp. 41–46.

[190] C. Kingsford and S. L. Salzberg, "What are decision trees?" *Nature biotechnology*, vol. 26, no. 9, pp. 1011–1013, 2008.

[191] M. Cosenza-Andraus, C. Nunes-Cosenza, R. Gomes-Nunes, C. Fantezia-Andraus, and S. Alves-Leon, "Video-electroencephalography prolonged monitoring in patients with ambulatory diagnosis of medically refractory temporal lobe epilepsy: application of fuzzy logic's model," *Revista de neurologia*, vol. 43, no. 1, pp. 7–14, 2006.

[192] E. B. Assi, D. K. Nguyen, S. Rihana, and M. Sawan, "Towards accurate prediction of epileptic seizures: A review," *Biomedical Signal Processing and Control*, vol. 34, pp. 144–157, 2017.

[193] A. K. Jain, "Data clustering: 50 years beyond k-means," *Pattern recognition letters*, vol. 31, no. 8, pp. 651–666, 2010.

[194] A. Malafeev, D. Laptev, S. Bauer, X. Omlin, A. Wierzbicka, A. Wichniak, W. Jernajczyk, R. Riener, J. Buhmann, and P. Achermann, "Automatic human sleep stage scoring using deep neural networks," *Frontiers in neuroscience*, vol. 12, p. 781, 2018.

[195] L. Vařeka, T. Prokop, R. Mouček, P. Mautner, and J. Štěbeták, "Application of stacked autoencoders to p300 experimental data," in *International Conference on Artificial Intelligence and Soft Computing*. Springer, 2017, pp. 187–198.

[196] J. Park and K.-E. Kim, "A pomdp approach to optimizing p300 speller bci paradigm," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 20, no. 4, pp. 584–594, 2012.

[197] Y. Zhang, P. Xu, K. Cheng, and D. Yao, "Multivariate synchronization index for frequency recognition of ssvep-based brain–computer interface," *Journal of neuroscience methods*, vol. 221, pp. 32–40, 2014.

[198] G. Pfurtscheller, C. Neuper, A. Schlogl, and K. Lugger, "Separability of eeg signals recorded during right and left motor imagery using adaptive autoregressive parameters," *IEEE transactions on Rehabilitation Engineering*, vol. 6, no. 3, pp. 316–325, 1998.

[199] S. Dharmasena, K. Lalitharathne, K. Dissanayake, A. Sampath, and A. Pasqual, "Online classification of imagined hand movement using a consumer grade eeg device," in *2013 IEEE 8th International Conference on Industrial and Information Systems*. IEEE, 2013, pp. 537–541.

[200] J. Decety, "Do imagined and executed actions share the same neural substrate?" *Cognitive Brain Research*, vol. 3, no. 2, pp. 87–93, 1996, mental representations of motor acts. [Online]. Available: https://www.sciencedirect.com/science/article/pii/092664109500033X

[201] S. Tayeb, A. Mahmoudi, F. Regragui, and M. M. Himmi, "Efficient detection of p300 using kernel pca and support vector machine," in *2014 second world conference on complex systems (WCCS)*. IEEE, 2014, pp. 17–22.

[202] M. H. Alomari, A. Samaha, and K. AlKamha, "Automated classification of L/R hand movement EEG signals using advanced feature extraction and machine learning," *CoRR*, vol. abs/1312.2877, 2013. [Online]. Available: http://arxiv.org/abs/1312.2877

[203] F. Wang, K. Kim, S. Wen, Y. Zhang, and C. Wu, "Eeg based automatic left-right hand movement classification," in *2012 24th Chinese Control and Decision Conference (CCDC)*, 2012, pp. 1469–1472.

[204] F. Flórez, J. M. Azorín, E. Iáñez, A. Úbeda, and E. Fernández, "Development of a low-cost svm-based spontaneous brain-computer interface." in *IJCCI (NCTA)*, 2011, pp. 415–421.

[205] J. Schmidhuber, S. Hochreiter *et al.*, "Long short-term memory," *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, 1997.

[206] G. Liu and J. Guo, "Bidirectional lstm with attention mechanism and convolutional layer for text classification," *Neurocomputing*, vol. 337, pp. 325–338, 2019.

[207] J. C. Nunez, R. Cabido, J. J. Pantrigo, A. S. Montemayor, and J. F. Velez, "Convolutional neural networks and long short-term memory for skeleton-based human activity and hand gesture recognition," *Pattern Recognition*, vol. 76, pp. 80–94, 2018.

[208] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[209] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*. PMLR, 2015, pp. 2048–2057.

[210] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, 2016, pp. 1480–1489.

[211] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," *Advances in neural information processing systems*, vol. 28, 2015.

[212] F. Riaz, A. Hassan, S. Rehman, I. K. Niazi, and K. Dremstrup, "Emd-based temporal and spectral features for the classification of eeg signals using supervised learning," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 1, pp. 28–35, 2015.

[213] Z. Jiao, X. Gao, Y. Wang, J. Li, and H. Xu, "Deep convolutional neural networks for mental load classification based on eeg data," *Pattern Recognition*, vol. 76, pp. 582–595, 2018.

[214] R. El-Lone, M. Hassan, A. Kabbara, and R. Hleiss, "Visual objects categorization using dense eeg: A preliminary study," in *2015 International Conference on Advances in Biomedical Engineering (ICABME)*. IEEE, 2015, pp. 115–118.

[215] A. Kapoor, P. Shenoy, and D. Tan, "Combining brain computer interfaces with vision for object categorization," in *2008 IEEE conference on computer vision and pattern recognition*. IEEE, 2008, pp. 1–8.

[216] C. Spampinato, S. Palazzo, I. Kavasidis, D. Giordano, N. Souly, and M. Shah, "Deep learning human mind for automated visual classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6809–6817.

[217] C. J. Ortiz-Echeverri, S. Salazar-Colores, J. Rodríguez-Reséndiz, and R. A. Gómez-Loenzo, "A new approach for motor imagery classification based on sorted blind source separation, continuous wavelet transform, and convolutional neural network," *Sensors*, vol. 19, no. 20, p. 4541, 2019.

[218] M. T. Sadiq, X. Yu, Z. Yuan, Z. Fan, A. U. Rehman, G. Li, and G. Xiao, "Motor imagery eeg signals classification based on mode amplitude and frequency components using empirical wavelet transform," *Ieee Access*, vol. 7, pp. 127 678–127 692, 2019.

[219] S. Taran and V. Bajaj, "Motor imagery tasks-based eeg signals classification using tunable-q wavelet transform," *Neural Computing and Applications*, vol. 31, no. 11, pp. 6925–6932, 2019.

[220] H. Dose, J. S. Møller, H. K. Iversen, and S. Puthusserypady, "An end-to-end deep learning approach to mi-eeg signal classification for bcis," *Expert Systems with Applications*, vol. 114, pp. 532–542, 2018.

[221] Y. Hou, L. Zhou, S. Jia, and X. Lun, "A novel approach of decoding eeg four-class motor imagery tasks via scout esi and cnn," *Journal of neural engineering*, vol. 17, no. 1, p. 016048, 2020.