Application of Deep Neural Networks in Electroencephalography (EEG): Classification of User Intention

Lakshmi Krisha Kanumuru

A Thesis submitted to the University of Kent
for the degree of
Doctor of Philosophy in Electronic Engineering

School of Engineering, University of Kent
Submitted in May 2022
Page Count: 201

Abstract

A primary topic of focus for Brain-Computer Interfaces (BCI) research is to accurately classify EEG signals associated with various motor imagery (MI) activities. Difficulties such as intra- and cross-subject variability, and restricted open-access data availability make establishing firm findings challenging. Deep learning approaches have gained widespread use in various application fields, including natural language processing and computer vision. We are interested in improving the performance of traditional classification methods in EEG-MI and the reliability and robustness of EEG-MI classification utilising deep networks.

This thesis presents two frameworks based on deep Convolutional Neural Networks (dCNN). The first is a spectrogram and dCNN-based method that provides the network with information about the EEG data's power spectrum. The second is a topographical map and dCNN-based method capable of retaining the spatial, temporal, and spectral information contained in the data fed to the dCNN.

The spectrogram based method with a 2-class MI classification problem performed with an average model accuracy of 91.81%, across individual participants. Results for the topographical maps based method demonstrate that it is capable of robustly and accurately identifying MI for two and three-class datasets and provides a feasible method for BCI applications employing EEG-MI. We concluded in this investigation that for two-class MI datasets the proposed topographic and dCNN framework was able to generalise across individuals in a dataset, with its best performance at 94.12%. For the three-class MI datasets the best performance was 98.48%.

As part of our future research, we intend to collect data from a variety of patients with varying cognitive abilities in order to further validate and refine the model. The use of active electrodes has shown promise in the literature; incorporating them into the data collection process may help to alleviate some of the shortcomings currently observed with inactive electrodes.

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my PhD supervisor, Dr Konstantinos Sirlantzis, for the continuous support and dedicated involvement in every step throughout the process of my PhD study and research. I would like to thank him for his for his patience, motivation, enthusiasm and immense knowledge.

A very special thank you to my undergraduate lecturers, Mr Winston Waller and Dr Peter Lee, for their guidance and moral support throughout the years.

I would certainly be remiss not to mention and sincerely thank the wonderful administration support staff in Jennison. Thank you to the technical support staff, who I've known since my undergraduate years - Antonio, Jo, Edwin and Yan. A special thank you to Tara, who helped me through the PhD application process, and who was just as excited as I was when I was awarded the Vice Chancellor's scholarship.

Most importantly, none of this could have happened without my family. I want to thank my wonderful parents, Soudhamini and Sridhar without the moral support of which I wouldn't have been able to get through this arduous journey. I extend my heartfelt thanks to my grandmothers, who offered their support through conversation and food, steadfastly supporting me. Additionally, I would like to extend thanks to my extended family for their support through my academic journey. This dissertation stands as a testament to my family's unconditional love and encouragement.

I would also like to thank the many friends and colleagues I have met throughout my time at Kent - Elakkiya, Julie, Pooja, Shabnam, Kenan, Robert, Luma, Hind, Estela, Philippos, Anthony, and Sumita. I cannot begin to express my gratitude and appreciation of their friendship. Additionally, I would like to thank my friends who have stuck with me from my undergraduate years - Tyler and Pooja.

I would like to express my gratitude to my co-workers from Mocketts. They have seen me

through many years of study and growth. Having since started a job in Manchester, I give special thanks to Sean, and Joy for their professional and personal support since the very beginning.

To Cristel, my soul sister, a pillar of strength, and who could always make me laugh on the darkest of days.

Finally, to Paul, my partner, best friend and closest confidant, without whom I couldn't have done any of this. I can never express the immensity of my gratitude. Thank you.

This is to certify that I am responsible for the work submitted in this thesis, that the original work is my own except as specified in acknowledgements or in footnotes, and that neither the thesis nor the original work contained therein has been submitted to this or any other institution for a degree.

_____

Lakshmi Krisha Kanumuru

Contents

Abstract **i**

List of Figures **ix**

List of Tables **xii**

Abbreviations **xv**

1 Introduction **1**

1.1 Purpose 1

1.2 Thesis Objectives 6

1.3 Contributions 8

1.4 Thesis Structure 8

2 Literature Review **10**

2.1 Introduction 11

2.2 Motor Imagery 14

2.3 Principles of a BCI System 16
2.4 Measuring Brain Activity 20
2.5 Introduction to Traditional Machine Learning 29
2.6 Traditional Machine Learning for MI BCI applications 30
2.7 Limitations of conventional EEG data analysis 31
2.8 Deep Learning 32
2.8.1 Convolutional Neural Networks 34
2.8.2 Recurrent Neural Networks 36
2.8.3 Hybrid Structures 38
2.9 Preprocessing 39
2.10 Feature Extraction Methods 41
2.11 Databases 44
2.11.1 BCI Competition III - Dataset IIIA 45
2.11.2 BCI Competition III - Dataset IVA 46
2.11.3 BCI Competition III - Dataset IVB 47
2.11.4 BCI Competition III - Dataset V 47
2.11.5 BCI Competition IV - Dataset I 48
2.11.6 BCI Competition IV - Dataset IIa 49
2.11.7 BCI Competition IV - Dataset IIb 49
2.11.8 H.Cho Two-Class Dataset (GigaDB) 51

3 Deep learning using EEG spectrograms for classification of user intention 52

3.1 Introduction 52

3.2 Data Preprocessing 56

3.3 Model Implementation 57

3.4 Analysis and Results 61

3.5 Conclusions 70

4 Deep learning using topographical EEG images for classification of user intention 72

4.1 Introduction 72

4.2 Data Preprocessing 74

4.3 dCNN Model Implementation 77

4.4 Performance Metrics 79

4.5 Analysis and Results 81

4.5.1 GigaDB dataset 83

4.5.2 BCI III dataset IVA 92

4.5.3 BCI IV dataset I 99

4.5.4 BCI III dataset IVB 105

4.5.5 BCI III dataset V 109

4.5.6 BCI III dataset IIIA 115

4.6 Conclusions 121

5 Concluding remarks and recommendations for Future work 124

5.1 Discussion 124

5.2 Challenges 128

5.2.1 Challenges in research and development 129

5.2.2 Challenges in commercialisation 130

5.2.3 Shortcomings in testing approaches 134

5.2.4 Issues preventing wide-spread adoption 135

5.2.5 Ethical considerations 135

5.3 Future Work 135

Bibliography 138

Appendices 158

Appendix A EEG Signal Artefacts 159

A.1 Physiological Artefacts 159

A.2 Non–Physiological Artefacts 160

Appendix B Experimental Setup 162

Appendix C Deep Learning Optimisation 164

C.1 Regularisation 165

C.2 Gradient Descent Variants 169

C.3 Gradient Descent Optimisation Algorithms 170

C.4 Network Structure 178

C.5 Additional Optimisation Strategies 182

C.6 Hyperparameters Selection 183

List of Figures

2.1 General architecture of BCI interface 19

2.2 Brain: Motor Cortex 21

2.3 Mu-rhythms 22

2.4 Types of EEG rhythms 24

2.5 Electrode types used in BCIs 25

2.6 10-20 Electrode Placement System 28

2.7 BCI Competition III - Dataset IIIa - 60 channel electrode positions 46

2.8 Timing paradigm of BCI III - Dataset IIIa 46

2.9 Acquisition of dataset IIb from the BCI Competition IV 50

3.1 EEG signal representations 55

3.2 MI Classification Pipeline using STFT and deep learning framework 56

3.3 Matrix of electrodes vs the time points 58

3.4 MI classification using STFT and deep learning framework 59

3.5 Spectrogram-dCNN Confusion Matrix: BCI IIIa binary classification 64

3.6 Model Accuracy: BCI IIIa binary classification 65

3.7 Spectrogram-dCNN Confusion Matrix: BCI IIIa 4 class 67

3.8 Model Accuracy: BCI IIIa 4 class classification 68

3.9 Model Loss: BCI IIIa 4 class classification 68

4.1 A sample of the produced 2–D azimuthal projections 77

4.2 Proposed dCNN Model 79

4.3 Confusion Matrix for proposed Model over GigaDB Dataset 84

4.4 Accuracy Curves for proposed Model over GigaDB Dataset 86

4.5 Loss Curves for proposed Model over GigaDB Dataset 87

4.6 ROC Curves for proposed Model over GigaDB Dataset 89

4.7 Confusion Matrix for proposed Model over BCI III Dataset IVa 93

4.8 Accuracy Curves for proposed Model over BCI III IVa Dataset 95

4.9 Loss Curves for proposed Model over BCI III Dataset IVa 96

4.10 ROC Curves for proposed Model over BCI III Dataset IVa 97

4.12 Accuracy Curves for proposed Model over BCI IV Dataset I 100

4.13 Loss Curves for proposed Model over BCI IV Dataset I 101

4.14 ROC Curves for proposed Model over BCI IV Dataset I 102

4.11 Confusion Matrix for proposed Model over BCI IV Dataset I 104

4.15 Confusion Matrix for proposed Model over BCI III Dataset IVb 106

4.16 ROC Curves for proposed Model over BCI III Dataset IVb 108

4.18 Accuracy Curves for proposed Model over BCI III Dataset V 110

4.19 Loss Curves for proposed Model over BCI III Dataset V 111

4.20 ROC Curves for proposed Model over BCI III Dataset V 112

4.17 Confusion Matrix for proposed Model over BCI III Dataset V 114

4.21 Confusion Matrix for proposed Model over BCI III Dataset IIIa 116

4.22 Accuracy Curves for proposed Model over BCI III Dataset IIIa 117

4.23 Loss Curves for proposed Model over BCI III Dataset IIIa 118

4.24 ROC Curves for proposed Model over BCI III Dataset IIIa 120

C.1 Fully connected complex network 167

C.2 Fully connected network with 50% dropout 168

C.3 Momentum update v/s Nesterov momentum update 171

C.4 Binary Step Function 178

C.5 Example of a Linear Activation Function: $y = 2x$ 179

C.6 Sigmoid Activation Function 180

C.7 Tanh Activation Function 181

C.8 ReLU Activation Function 181

C.9 Leaky Rectified Linear Unit 182

C.10 Visual representation of types of hyperparameters search 184

List of Tables

2.1 MI-EEG Classification techniques using traditional Machine Learning 16

2.2 Brain wave categorisation, frequency ranges and general attribution 22

2.3 Properties of various signal acquisition systems 26

2.4 List of Physiological and Non-physiological artefacts 41

2.5 Detailed open-source EEG datasets 45

3.1 Comparative results of two-class BCI competition III dataset IIIa 66

3.2 Comparative results of four-class BCI competition III dataset IIIa 69

3.3 Comparative results of two-class BCI competition III dataset IIIa 70

4.1 Talos Hyperparameter search 82

4.2 K-Fold summary for dataset GigaDB 90

4.3 Resnet50 K-Fold summary for dataset GigaDB 91

4.4 K-Fold summary for BCI III dataset IVa 98

4.5 Resnet50 K-Fold summary for BCI III dataset IVa 98

4.6 K-Fold summary for BCI IV dataset I 103

4.7 Resnet50 K-Fold summary for BCI IV dataset I 103

4.8 K-Fold summary for BCI III dataset IVb 107

4.9 Resnet50 K-Fold summary for BCI III dataset IVb 107

4.10 K-Fold summary for dataset BCI III Dataset V 113

4.11 Resnet50 K-Fold summary for dataset BCI III Dataset V 113

4.12 K-Fold summary for BCI III Dataset IIIa 119

4.13 Resnet50 K-Fold summary for BCI III Dataset IIIa 119

Abbreviations

AI	Artificial Intelligence
ANN	Artificial Neural Network
AUC	Area Under the Curve
BCI	Brain Computer Interface
CNN	Convolutional Neural Network
CSP	Common Spatial Pattern
CWT	Continuous Wavelet Transform
dCNN	Deep Convolutional Neural Network
DFT	Discrete Fourier Transform
DL	Deep Learning
DNN	Deep Neural Network
ECoG	Electrocorticography
EEG	Electroencephalography
ERD	Event Related Desynchronisation
ERP	Event Related Potential
ERS	Event Related Synchronisation
FFT	Fast Fourier Transform
fMRI	Functional Magnetic Resonance Imaging
FPR	False Positive Rate
ICA	Independant Component Analysis

KNN	K-Nearest Neighbour
LDA	Linear Discriminant Analysis
LSTM	Long Short-Term Memory
MI	Motor Imagery
ML	Machine Learning
MLP	Multi-layer Perceptron
NN	Neural Networks
PCA	Principal Component Analysis
PSD	Power Spectral Density
ReLU	Rectified Linear Unit
ROC	Receiver Operating Characteristic
SGD	Stochastic Gradiant Descent
SSVEP	Steady State Visually Evoked Potential
STFT	Short-term Fourier Transform
SVM	Support Vector Machine
TPR	True Positive Rate
VEP	Visually Evoked Potential
WHO	World Health Organisation
WT	Wavelet Transform

1

Introduction

1.1 Purpose

The World Health Organisation (WHO) reported in 2011 that more than one billion people worldwide suffer from some form of disability. Of those, nearly 200 million experience difficulties in functioning. The 2011 WHO report explained that disability is on the rise due to the increased ageing populations and those with chronic health conditions which may have an associated disability [1]. The increase, as mentioned in [1], is reflected in the survey conducted in the UK between 2019 and 2020, where 14.1 million people reported having a disability, up 2.7 million from 2009 and 2010, when 11.4 million individuals reported having a disability. The percentage of people who reported a disability climbed to 22% in 2019-2020, up from 19% in 2009-2010 [2].

Mobility limitations are the leading cause of functional limitations in adults. Between 2014 and 2015, in the United Kingdom, it was recorded that mobility-related disabilities accounted for 53% of the total reported instances. Though disability has no gender bias, the possibility of developing a disability increases with age. Within the general population, 45% of adults

above pension age have a disability. For working-age adults, this figure sits at 16% and for children at 6%. Only 17% of the total persons have had a disability from birth [3]. Mobility difficulties are also strong predictors for those who may have difficulty with activities of daily living. A variety of neurological issues such as motor neuron disease, stroke, spinal cord injury or acute brain injuries may lead to some form of paralysis. This may deny them the possibility of using conventional assistive devices for functional movements, such as a joystick to control a wheelchair or exoskeleton.

From the synthesis of personal computing in the 1970s, researchers have endeavoured to narrow the gap between humans and computer technology. The process initially started with the development of graphical user interfaces and computers [4]. Such individuals would benefit from a Brain-Computer Interface (BCI) solution and significantly boost their autonomy and ability to interact with the world. BCI represents a communication relay between a human and a computer - it offers the subject a means of interacting with the surrounding environment by interpreting signals generated by a person's brain.

The first BCI was suggested in 1964 by Dr Grey Walter. He conducted an experiment where he found that by connecting electrodes to the brain's control centres (which he had previously found through another investigation), i.e. the motor cortex, the patient in question could control a slide show projector to flip to the following image. BCI research was limited at the time, but the importance has since grown to be known, having radically expanded. The applications of BCI have since grown beyond just being a tool of communication and control for those who cannot otherwise communicate; one of the more popular utilisation is hands-free gaming [5]. This is why the idea of BCI based assistive control has been of great interest, as it requires no muscular movement, and the directional commands rely on the electrical activity generated by the brain.

BCI systems consist of six stages: brain activity measurement, pre-processing, feature extraction, classification, translation into commands and feedback. The work presented in this PhD thesis belongs to the framework of BCI research. It focuses on the research of EEG data

processing and classification techniques to create and implement BCI for engaging with assistive technologies.

BCI techniques have various applications and can vary significantly in the level of invasiveness of the electrodes. Highly invasive techniques imply the insertion of electrodes into or just above the cortex through neurosurgery. Such measures imply a high level of risk and generally only find applications in scientific research and for medical purposes, such as deep brain stimulation for managing Parkinson's disease or severe epilepsy.

Non-invasive BCI techniques imply acquiring signals from the brain through electrodes placed on the scalp, such as Electroencephalography (EEG), to detect electromagnetic waves created by the brain or using functional magnetic resonance imaging (fMRI) by monitoring blood flow in different regions of the brain. The vast majority of studies have focused on EEG based BCI interfaces.

Electroencephalography is one of the most widely used methods for recording the electrical activity of the brain. The analysis of physiological signals such as EEG has been helpful in the diagnosis of brain illnesses such as epilepsy, dementia, stroke, and brain tumours [6]. EEG is a popular choice for use in BCIs due to its high resolution, relative ease of use, portability, and cost-effectiveness compared to other methods. However, EEG signals pose challenges whilst processing. Due to their static nature, they are prone to suffering from external noise and signal artefacts. EEG signals, the electrical activity measured from the scalp, are also affected by mood and even the posture of a subject [7].

Due to advancements in EEG headset technology, it is now possible to gather EEG data more readily. EEG sensor technology is available at a low cost, with products developed by the entertainment industry, but also aimed at health and well-being, and education [8]. Although BCI can be designed to use EEG signals in a wide variety of ways, MI-BCIs have been subject to extensive research [9]. Subjects visualise movements occurring to their limbs to control a system using this MI-BCI. This interest is due to their broad applicability in areas such as neurorehabilitation and gaming, where the decoding of subjects thoughts of

imagined movement would be extremely useful [10, 11]. MI EEG–based BCI is a promising technology due to the multitude of use cases in medical and non–medical areas.

The MI-based BCI application entails the real-time interpretation of EEG data and the determination of responses to such signals. The curse of dimensionality problem [12] arose from non-stationary signals, multi-channel recording paradigm, channel correlation, and the presence of noise and artefacts while analysing EEG signals. As a result, robust and fast systems are needed to extract latent variables from physiological inputs. The implementation of MI EEG signals in real-world systems, in particular, is fraught with difficulties [13].

Artificial Intelligence (AI) has shown considerable promise in interpreting and analysing EEG data due to significant computational capacity advances. EEG data refers to the electrical activity measured from the scalp that reflects the activity of neurons in the brain. Machine Learning (ML), a subset of AI, is the ability of a complex algorithm to learn from provided data without being explicitly programmed [14]. For a long time, researchers have employed conventional neural networks (NNs) to automate various tasks. With the exponential growth of all digital data kinds (time-series signals, pictures, and videos), however, it is quickly apparent that NNs are insufficient to cover all stages of variability within a vast amount of data [15]. Deep learning (DL) techniques for evaluating large-scale datasets have recently sparked a revolution [13]. Whereas, unlike standard NNs, DL approaches can learn a vast number of the essential parameters for extracting meaningful and in-depth information. Furthermore, DL does not rely on time-consuming, subject-dependent [16] hand-engineered techniques of extracting features, where you may suffer from loss of information.

Deep neural networks (DNNs) require a considerable number of learn-able parameters and intensive floating-point matrix multiplication [17, 18, 19]. The computing system is put under much strain as a result of this. Typically, the research incorporates graphical processing units (GPUs) to tackle the problem of heavy computation. DNNs are trained by instances,

necessitating a big enough training dataset with various examples that are not biased to some instances. The lack of large enough datasets is the main challenge that prevents researchers from applying DL using MI EEG signals. The existing MI EEG datasets, on the other hand, are recorded for multiple people and only comprise a small number of MI trials. MI trials refer to a type of experimental task used in BCI research.

In particular, Convolutional Neural Networks (CNN) are highly effective at detecting complex features and patterns in images for image classification. Innately complex properties and non-linear interactions between these variables can be modelled inherently by the deep network, composed of multiple layers of neurons. Given that brain patterns in EEG are complex and non-linear, DL approaches can achieve a greater success rate in the classification of these patterns than traditional methods. Aside from that, constant data recording in BCI systems generates a large amount of data crucial in the training of DNNs. EEG signal acquisition is used to examine BCI systems in this study, which is the first of its kind to do so. The primary goal of this thesis is to investigate the signal processing component of BCI. Specifically, MI signals are used in this study in order to control the BCI system.

DL algorithms were researched for the classification of brain patterns associated with MI. BCI signals are acquired, processed, and fed back to the user through EEG signals; this is accomplished through a series of studies in which MI is used to control BCI, and DL methods are utilised for signal processing. Examples are required by learning algorithms in order for them to infer the underlying statistical structure of the respective brain state. To achieve a specific brain state repeatedly throughout a calibration session, subjects are required to perform the following tasks: Current learning machines are capable of extracting spatio-temporal blueprints of these brain states from even little quantities of data, which may then be used in the subsequent feedback session to improve the accuracy of the predictions.

A key issue in BCI research is dealing with the enormous variation from trial to experiment. This endeavour requires the use of advanced ML techniques such as deep learning, which are critical in this process. It is possible to achieve high decision accuracy for BCI by utilising

state-of-the-art learning machines; however the classification process produces a by-product that identifies the few most salient traits for classification, which may then be paired with neurophysiological information. Therefore, ML approaches are valuable beyond the pure classification or adaptive spatio-temporal filtering stage, as they can contribute to a better interpretation and understanding of a novel paradigm that is being studied.

1.2 Thesis Objectives

The focuses of this thesis is on investigating the improvements in accuracy and robustness that can be made to state–the–art DL methodologies. Recent successes in image classification have used dCNNs designed to exploit invariance and capture compositional features in data. For the most part, these systems were equipped to handle image data or audio data. This research explores if we are able to accurately and robustly classify EEG signals associated with various MI activities.

There are two principal aims of this research. Firstly, the improvement on results from classic EEG MI signal classification techniques using deep learning methodologies. Secondly, improving the reliability and robustness of MI EEG signal classification using DL networks. This thesis explored the use of EEG data representations as inputs to deep networks for MI classification. It also explored the possibility of a model which could generalise over multiple users in a dataset, as opposed to the current literature which aims to classify on individual subjects.

The author was motivated to empower users who are unable to use conventional assistive technology control inputs. By aiding users with motor impairments in training and rehabilitation and interacting with smart technology solutions to improve their quality of life, and diversify the way we interact with technology. Through two studies, this research aims to address the challenges in classifying EEG images. The experiments explore spatial, temporal and frequency variants of the data in order to provide the most suitable format which

can be appropriately trained by the model.

Two principal approaches were explored to attempt and address the research questions. The first proposed approach aims to use generated spectrograms as an input to a CNN architecture for MI classification. The second approach explored a classification pipeline for modelling user intentional events from multi-channel EEG time–series data. The proposed approach preserves the spatial, temporal and spectral structure of EEG. One of the challenges of modelling cognitive events from EEG is finding representations that are invariant to differences in one particular subject's data, and also between subjects, and the noise associated with data collection. An overview of the two approaches and their respective research questions is summarised as follows:

Study 1: Application of deep CNN networks in the interpretation and analysis of EEG MI signals represented as windowed spectrogram images. The salient research questions of the study are as follow:

1. Would providing a spectrogram, frequency, representation of EEG signals to a deep network improve on results obtained from classical ML and state–of–the–art methodologies?
2. Is a frequency only representation enough to extract relevant features by the deep model?

Study 2: Application of deep networks in the interpretation and analysis of EEG MI signals represented as topographical maps. The salient research questions of the study are as follow:

1. Would providing a topographical representation of MI EEG signals to a deep network improve on results obtained from classical ML using state–of–the–art methodologies?
2. Are we able to find a robust model which is able to generalise over all the subjects, and would not be required to train on an individual subject-to-subject basis?

The open-source datasets chosen for these experiments will be discussed further in Chapter 2, section 2.11. These have been selected due to their popularity for benchmarking analysis and have become well-used within this research area. Thus, we are able to compare results between given data pipelines and models accurately.

1.3 Contributions

There were three main contributions of this thesis: Firstly, the development of a novel and robust dCNN network which has the ability to generalise over multiple subjects. It has the ability to classify MI EEG data, where two class models achieved up to 95% accuracy. The model achieved results which were on par with state-of-the-art models. Secondly, the transformation of raw data to topographical and spectral representations as an input to the created dCNN network. Thirdly, the creation of a novel classification pipeline and subsequently feeding into a DNN, which showed improvements when compared to state-of-the-art deep learning models.

1.4 Thesis Structure

By addressing the difficulties inherent in the processing and interpreting EEG data containing MI, this research seeks to improve on conventional ML approaches and state-of-the-art DL procedures.

As a starting point, the Literature Review (Chapter 2) discusses the current state of the literature on EEG MI data, analysis approaches, BCIs and their applications in healthcare, and the difficulties that researchers encounter when analysing such data. We discuss machine learning in the context of classifying EEG data. DL is introduced along with its significance in the healthcare industry and in the field of research into EEG classification. This chapter

discusses in detail the state-of-the-art strategies for analysing EEG MI data that employ deep learning techniques.

In the Spectrograms EEG representations chapter (Chapter 3), we examine a DL framework inspired by recent promising results for image classification. We use dCNNs to exploit invariance and capture compositional elements in EEG signals. Historically, dCNNs have been largely employed to manage image or audio data. An objective of our research is to apply dCNNs on EEG spectrograms, to correctly classify user intentions. The methodology effectively treats EEG channel data as an audio representation, and the suggested method is modelled after analogous applications using DNNs in that domain. Throughout the course of this study, several DNN models were explored.

A different approach explored through our research is discussed in the Topographical EEG Representations chapter (Chapter 4). This consists of transforming EEG signals into topographical images which are then used as inputs to a deep network. These representations are subjected to analysis by the trained DNN model, and we investigate the experimental results over a series of datasets. Finally, we conclude which framework and model work best against the given topographical representations. We explain the results obtained for both individual subjects and as a generalised model across the entire dataset.

In the Concluding Remarks and Recommendations for Future Work (Chapter 5) we summarise the work done throughout this investigation and bring the findings to a close in Chapter 5. The difficulties in research and development, the shortcomings of testing methodologies, ethical considerations, and the issues that limit widespread adoption and commercialisation are further explored. Additionally, work beyond the conclusion of this research is explored, emphasising the creation of improved DL methods and alternative approaches for the classification of EEG data in the future.

2

Literature Review

This chapter explores the current literature and concepts used within the course of this research. Previous works in the fields of electrode systems for recording EEG signals, MI as a classification tool and the current state of the art in these areas of research are explored.

First, sections 2.2, 2.3 and 2.4 provide an overview of the MI, principals of a BCI system and measuring brain activity, respectively. Further to this, sections 2.5 and 2.6 cover the introduction to traditional ML and the applications of MI EEG in current literature. We also explore the limitations of conventional ML and how these can be addressed with the use of DL methodologies. Section 2.8 provide a background in the concepts of DL, NNs, and the related MI-EEG background in the state-of-the-art. Section 2.2 describes the current state of the art in MI based systems. Section 2.9 provides an overview of the current pre-processing techniques which are used within literature when handling EEG signals. Finally, we introduce the open-source MI-EEG datasets which were used during the course of this research in Section 2.11.

2.1 Introduction

EEG is a non-invasive clinical method that neurologists and physicians have routinely utilised for decades to diagnose brain-related illnesses such as epilepsy and seizures [20]. However, manually annotating clinical data is time-consuming and expensive, requiring highly qualified board-certified neurologists in short supply. It is known to be quite a tiresome task [20].

EEG signal interpretation is quite subtle and difficult to decipher, thus making human errors probable [21]. Much of the information acquired is now available in digital format due to recent advancements in EEG technology. However, there are just a handful of publicly accessible datasets. While acquiring EEG data for processing remains challenging, the amount of data available has steadily increased. The advancement of technology necessitates hiring trained experts to read the data, hence increasing the possibility of inaccuracy [22].

Automated EEG data analysis can assist humans in providing intelligent solutions via the use of AI, which can be aided by technological advancement. AI can engage in human-like thought patterns, allowing it to learn, reason, and self-correct the same way humans do [23]. Davenport [24] divided AI into three different subgroups: process automation, cognitive insight and cognitive engagement. Process automation is the automation of digital and physical tasks. Cognitive insight is using automation algorithms, such as ML algorithms, to detect patterns in large data and interpret their meaning. Finally, cognitive engagement is used in projects which engage using natural language processes.

In recent years, ML has been applied to EEG data as a subcategory of AI. On the other hand, conventional ML techniques are constrained by the requirement to extract features to train a model manually. This time-consuming process requires prior knowledge of the domain from which the features are being extracted, among other requirements.

A more recent addition in the field of AI comes in the form of DL. Because of advances in technology, it has gained significant traction and attention in the last decade. An advantage

of DL is that it allows for automated data analysis to be performed, and features do not need to be manually extracted before being passed to the model. On the other hand, the training data must be of high quality and annotated appropriately. Both ML and DL have been successful, given an abundance of data, such as object identification and voice recognition, and both have been successful in other situations [25]. However, due to a scarcity of freely available EEG data in the research sector, its application in this context was limited.

EEG data can be utilised for a variety of different purposes, including diagnosis and treatment in medicine. It can be used to aid in clinical diagnosis via the increasingly prevalent use of brain imaging techniques. Through the use of activation characteristics research, it is possible to observe the engagement of a specific brain region and the involvement of physiological or cognitive activity. Analysed EEG data can be used to ascertain the characteristics of a person's brain state, such as whether they are acting willingly or involuntarily, first to target specific BCI applications [26].

EEG data can be processed in the time domain, frequency domain, or spatial domain, depending on the approach used. We sought to investigate EEG data in both the frequency and spatial domains in this research. The most natural way to handle EEG data is to take a time-domain waveform of the multi-channel signal. This is the method via which the majority of EEG acquisition equipment stores data. However, deriving usable information directly from waveform data is generally impractical. Despite almost two decades of research towards automatic classifiers, qualified technicians may discover anomalies directly from raw EEG waveforms in the medical industry [27].

There is BCI-based research that analyses time-domain data. In this field, the use of common spatial pattern (CSP) filters was pioneered. Using simultaneous diagonalisation of two matrices, this approach partitioned the signal into subcomponents with the greatest variance differences [28]. This has achieved well–recognised results, especially in [29, 30] which further introduced more methodologies. Additionally, CSP inspired EEGNet, a modern DNN-based classifier. According to the article, one advantage of processing data directly in its

raw form is avoiding the computational overhead associated with data transformation into a different domain or format [31].

EEG analysis for research purposes is usually performed in the frequency domain. Oscillations along the time axis reflect underlying brain activity and noise introduced by artefacts and other unforeseen factors. By sacrificing temporal information, we can extrapolate certain frequency distribution and power spectrum features that are more prominent than those of time-domain wave bursts and oscillations. When data is transformed from the time domain to the frequency domain, data with unusual variance within the time domain can be discarded. As a result, it is possible to develop significantly more sophisticated biomarkers.

Numerous physiological and cognitive breakthroughs have been made due to event-related potential (ERP) and event-related spectral perturbation research [32]. The time-domain to frequency-domain transformation is typically accomplished using the discrete Fourier transform (DFT), the fast Fourier transform (FFT), or wavelet transforms. Wavelet analysis frequently analyses both the frequency and time scales, implying that it is not a purely frequency-oriented methodology. Specific frequency bands are of particular interest in current research to ascertain the relationship between specific brain activities or functionalities [32, 33, 34, 35, 36].

In order to suppress artefacts and increase the signal–to–noise ratio, pre–processing of the EEG signals is general practice before further processing the EEG data A.2. As mentioned previously, there are a limited amount of available datasets, even more so, there are an even smaller amount publicly available MI EEG recordings.

Within literature, there are two popular techniques which have been applied attempting to increase the performance accuracy of these datasets. First, using a transformation state before the deep network structure. Techniques such as CSP [29], FFT [33], STFT [37] or CWT [38] have been widely examined. This has been observed to be the preferred methodology for researchers. The second option is transfer learning [39], however this required a large network to be pre–trained using another dataset before being re–trained through MI EEG

dataset.

2.2 Motor Imagery

Humans have the ability to imagine anything if they have any experience related to - an action, an object or a scenario. All movement by the human body which occurs voluntarily begins as thought–based on intention. Some may have the ability to imagine actions, but may not have the physical ability to action them.

MI is the cognitive process where a person imagines a motor action without executing the movement in question. This process requires the activation of certain areas of the brain which are dedicated to the preparation and execution of movement - the motor cortex. It has been widely accepted that similar areas are active during the mental imagery or physical movement of the same motion. MI trials are a type of experimental task used in research of BCI. Imagining motions of the right hand, left hand, right foot, left foot, both feet, and tongue are the most commonly employed MI activities in research. Many other tasks, such as those related to the elbow, fists, and fingers, are also investigated. The purpose of these trials is to observe changes in brain activity patterns as a result of the mental imagery task, which can then be used to control BCI devices or for other purposes. The EEG signals recorded during motor imagery trials can provide information about the neural activity associated with specific types of movements. Due to this there are variations in the motor cortex region of the brain (labelled in Fig. 2.2). Variations that are seen can include the increase or decrease of power spectrum or amplitude of produced EEG signals. By interpreting EEG signals generated from MI related tasks, we can deduce the intention of movement or MI activity patterns.

Once this electrical impulse is generated, the signal is passed through the nervous system to the intended body part with instructions of movement. The exact process occurs during involuntary action, such as heart rate, blood pressure and breathing [40]. However, the

signal for these particular actions is not produced in the motor cortex but the medulla oblongata [41]. Transient increases and drops in spectral power recorded in the human EEG have traditionally been referred to as event-related synchronisation (ERS) and event-related desynchronisation (ERD) [42], can be used to discern different imagined movements. Simply put, ERS and ERD occur when a user imagines moving a unilateral limb, with the recorded signals being amplified at the contralateral motor-sensory cortex and decreased at the ipsilateral. Both phenomena are time-locked to the event rather than phase-locked and are very frequency band-specific.

It has long been recognised that movement alters the frequency of the EEG [43]. Alpha and Beta bands may detect both synchronisation and desynchronisation [44]. For example, when the right or left hand is imagined, a similar ERD can be found over the contralateral hand area and an ERS over the ipsilateral hand area. The ERD phenomena serve as the basis for classifying MI. MI EEG signals are highly variable [45] due to physiological and psychological factors for each person at each given moment. This variation in recorded signals can be interpreted as a minor shift in the frequency of ERS/ERD. As a result, utilising conventional signal processing techniques to hand-engineer features would generate subject-dependent features, culminating in a BCI system that is not universal.

Traditionally MI EEG signals have been classified using ML techniques, and in earlier studies, standard NNs have been used to classify MI Signals. Table 2.1 shows the classification results observed by traditional ML techniques. In recent times, DNNs have made substantial progress in achieving high performance in classification tasks [46].

In order to trigger an MI EEG signal within an experiment environment, a subject must imagine performing a directed physical activity, such as the movement of a limb [58]. During data collection, a visual prompt is given to the subject to perform this imagination activity. This time–point is typically indicated within the data to mark the beginning and end of the MI activity in question. MI EEG signals are known to play a significant role within BCI technology [13].

Table 2.1: MI-EEG Classification techniques in literature using traditional Machine Learning

Methods	Accuracy %	No. of Classes	Classifier	Dataset
Alansari's [47]	83.8	2	SVM	BCI IV-IIb
Aljalal's [48]	80.2	2	MLP	BCI III-IVa
	89.4			
Behri's [49]	94.5	2	SVM KNN MLP	BCI III-IVa
	67.4			
Mirnaziri's [50]	61.7	4	MLP	BCI IV-IIa
Mishuhina's [51]	89.8	4	LDA	BCI III-IIIa
Li's [52]	68.6	3	SVM	BCI IV-IIa
	92.2			
Molla's [53]	91.2	2	SVM	BCI III-IVa
	92.2			
Silva's [54]	67.8	2	MLP	BCI IV-IIb
Wang's [55]	77.2	4	MLP	BCI III-IIIa
Wang's [56]	81.2	2	SVM	BCI IV-IIb
Zhang's [57]	84	2	SVM	BCI III-IVa

2.3 Principles of a BCI System

A BCI is an interface that has the ability to interact with an external device and collect data in real-time. It has the ability to communicate and translate these brain signals collected between the user and the computer system. This field of research is relatively new, and has been on the rise for the past two decades. BCIs bypass the requirement for physical output and apply advanced algorithms to translate brain signals into action [59]. This section provides a background to BCI technology and the application of EEG signals with this area.

The brain is the command base of the nervous system - it has more than 100 billion neurons [60]. The motor cortex of the brain is where movement electrical impulses are seen. In particular, the role of the primary motor cortex is to generate these nerve impulses to execute some movement. It is only in the past few decades that research into BCI has become of greater interest. Prashant et al. discuss that there are four main factors which have contributed to this [61]. Firstly, the benefit for users who have sustained physical damage or have a condition that has left them completely immobile. Secondly, the developments in

understanding the nature of brain signals and their complex functionalities. Thirdly, the rise in low-cost microelectronics which have allowed for tasks to be performed through embedded systems and BCI. The final factor has been the recent growth in the research of complex ML algorithms.

Creating a BCI is a complex and time-consuming endeavour that requires multidisciplinary expertise in computer science, signal processing, neuroscience, and psychology. Two steps are normally necessary to operate a BCI. First, there is an offline training phase, followed by an online phase in which the BCI is used to recognise mental states, calibrate the system, and translate them into computer commands.

EEG-based BCI systems can be classified into two main categories: evoked and spontaneous. An evoked EEG refers to brain activity that is elicited in response to a specific stimulus, such as a flashing light or a sound. For example, a visual evoked potential EEG would measure the brain's response to a visual stimulus, such as a flashing light, to assess the function of the visual system. On the other hand, spontaneous EEG refers to the continuous electrical activity of the brain in the absence of any specific stimulation. For example, a resting state EEG measures the ongoing brain activity when a person is simply lying still with their eyes closed. As their names suggest, the evoked systems require external stimulation, whereas the spontaneous systems do not. Visual, auditory, and sensory stimulations can all be used in evoked systems. The two primary kinds of evoked systems are visually evoked potential (VEP) and ERP, and both are used in various applications. We sought to investigate EEG data in both the frequency and spatial domains in this research. Because they are generated based on a user's conception of accomplishing a task, MI signals are of the spontaneous type [62].

Compared to spontaneous systems, evoked systems have higher throughput [13], require less training, and can be mastered by a wider number of users. This is because evoked EEG measures the brain's response to a specific, well-defined stimulus. This allows for a clear and consistent measurement of the brain's activity, which can then be used to control a BCI

system. In contrast, spontaneous EEG measures ongoing brain activity, which can be more difficult to interpret and control due to the presence of background noise and other sources of electrical activity in the brain. Additionally, evoked EEG allows for the use of averaging techniques, which can enhance the signal-to-noise ratio and increase the reliability of the brain activity measurement. This results in a more accurate and consistent signal that can be used to control a BCI system, which in turn leads to higher throughput. Although the downside of evoked systems is that they require the attention of the subject to remain fixed on the stimuli at all times, which can be mentally exhausting [10, 63, 64].

People who have motor difficulties can use BCI to control their environment. These individuals can improve their quality of life by being able to do their day–to–day activities such as controlling their television, lights, and room temperature. In addition to Locomotion, a BCI application that assists disabled individuals with transportation is available. People with physical disabilities would be able to manoeuvre their wheelchairs independently in this approach [65].

Due to advancements in BCI technology, it is now possible to use BCIs for mass consumption. BCI introduces a new mode of interaction for those who play video games or work with computers. Several recent studies have discovered that simple video games such as Pacman could be controlled using motor imagery signals [66].

Speech communication, often known as silent speech, is one of the most common uses of the BCI for people who have communication problems, though it is also one of the most computationally expensive. A large number of speech studies have been conducted on this topic. In such experiments, subjects were asked to choose the target letter from an on-screen display. A variety of brain processes have been used in conjunction with a variety of stimuli.

P300 event-related brain potentials in communication applications are one of the most widely used paradigms for BCI control in communication applications. A large number of speech communication experiments in BCI make use of these signals. Other control signals, such as Steady-State Visual Evoked Potentials (SSVEP), have been employed extensively for voice

communication [67].

A closed-loop system procedure must be followed by an online BCI, which consists of three overall modules [68] with a total of six steps as mentioned in Section 1.1 and seen in 2.1. In the next few paragraphs each element of the BCI system is described in detail.

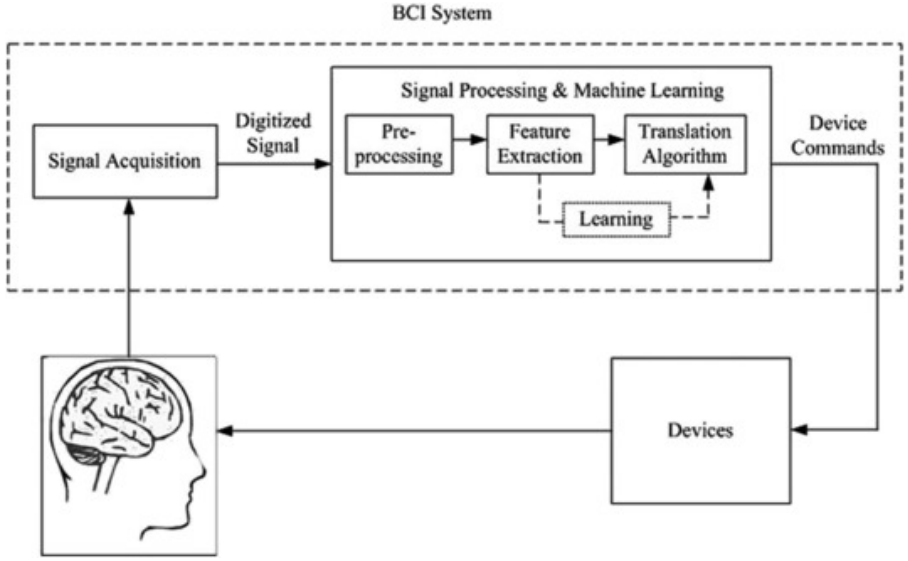


Figure 2.1: General architecture of BCI interface [69]

The signal acquisition module is responsible for recording the EEG signals, which provide the input for the BCI. Depending on the type of electrode (invasive or non–invasive), the signals are either recorded from the scalp and sit directly on the brain's surface to use neuronal activity. Signals are then amplified to enhance the signal strength and subsequently digitised before the pre-processing stage [68].

The preprocessing stage involves cleaning and de-noising input data to enhance the signal's relevant information. Raw EEG signals recordings, in addition to brain signals, also pick up muscular activity, eye movements and other such activities known as artefacts. Within BCI research, preprocessing is particularly of importance to obtain high classification accuracy [70].

The second task in the signal processing module includes three elements - feature extraction,

feature selection and classification. This module takes in the signals post preprocessing and is fed into one or more feature extraction methods. This component extracts features in the frequency and time domain. There are a variety of feature extraction methods that are used in BCI systems. This includes, but is not limited to: amplitude measures, band power, autoregressive models, wavelets, FFT, CSP etc., as mentioned in Section 2. A set of features derived from the signals is assigned a class in the classification stage [71]. This category corresponds to the type of mental state that has been identified. This process is also known as "feature translation" [72].

The processed EEG data undergoes feature extraction, feature selection, and feature classification phases to attain a decision. Manually selected, statistical, and data-driven adaptive features are the three types of EEG features commonly utilised in BCI systems, according to [73]. Several standard signal processing technologies were heavily used during these three phases. Given the dynamic and complicated nature of EEG signals [74], choosing a set of tools to deal with features is critical.

The final module is the application interface. The output device for most modern BCIs is a computer screen, and the output is a selection of targets, letters, or icons displayed on it [75]. Certain BCIs generate an output, such as the cursor moving toward the item before selecting it. The output device generates feedback to inform the user about the recognised brain activity pattern. This pattern is then used to maintain and improve communication accuracy and speed.

2.4 Measuring Brain Activity

The brain plays a role is most major body systems, including processing sensory information, regulating breathing and blood pressure. The cerebrum is the largest part of the brain and is divided into two halves called the hemispheres. It can be categorised into broad, functional areas: frontal, parietal, temporal and occipital.

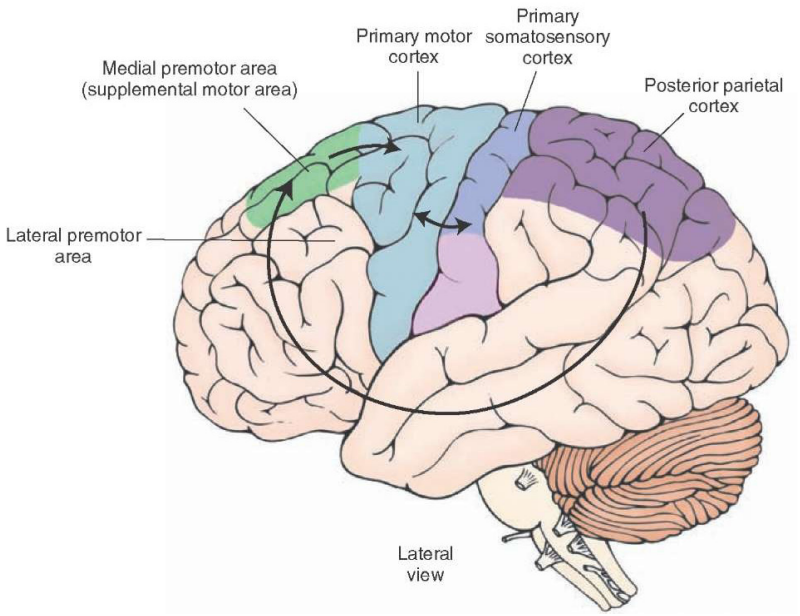


Figure 2.2: Brain: Motor Cortex [76]

The frontal lobe are the largest lobes of the cerebrum and are located near the front of the brain. This region is associated with reasoning, high–level cognition, and motor skills. Near the rear of the frontal region is the motor cortex. The primary motor cortex (M1), (Fig. 2.2, is responsible for the generation of nerve impulses that regulate movement. M1 signals propagate across the mid line, activating muscles on the opposing side. The primary somatosensory cortex somatotopically represents each part of the body. The amount of brain matter dedicated to a particular body region is proportional to the extent to which the primary motor cortex controls that region.

The parietal lobe is located in the mid–region of the brain and is associated with sensory information such as pressure, touch and pain. A portion of this region is known as the somatosensory cortex, which is essential to process the body's senses. The temporal lobe is located at the base of the brain, this lobe is responsible for interpreting the sounds and speech we hear. The hippocampus, associated with the formation of memories, is also located in the temporal region. The occipital lobe is located at the back of the brain and is associated with visual information and stimuli interpretation.

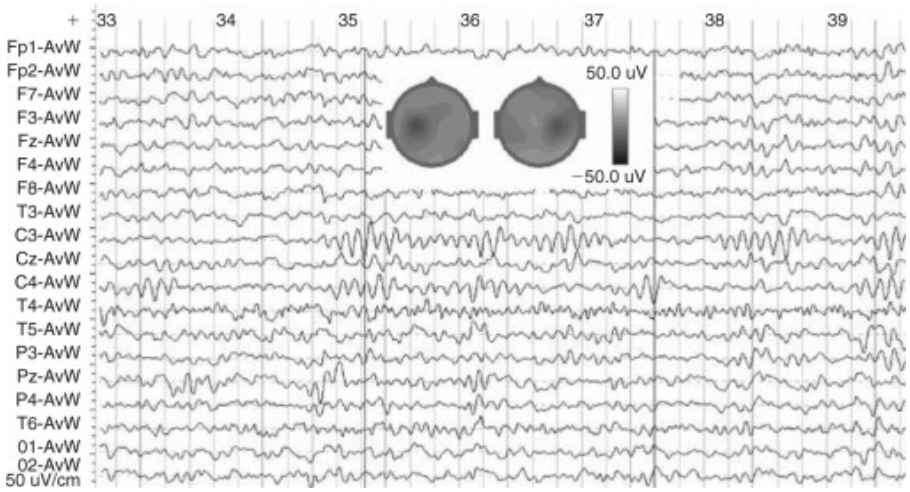


Figure 2.3: Mu-rhythms. A fragment of EEG recorded in a healthy adult subject during eyes open condition. Note distinct rhythms of about 10 cycles per second at C3 and C4. Insertion – two maps of potentials taken at two moments indicated bay black vertical lines.

EEG signals do not have distinctive waveforms such that of electrocardiogram or heart sounds. On the other hand, brain patterns generate wave shapes that are typically sinusoidal in character [77]. The measured EEG brain waves can be categorised depending on their frequency ranges (brain waves per second) and its amplitude (the voltage). A sample of an EEG recording can be seen in 2.3. Once this information passes through a Fourier transform, it transforms to the frequency domain, and power spectrum of the EEG signal is derived [77]. A summary of the various brain waves is shown below in Table 2.2. Figure 2.4 shows an example signal representation of the main brain signals.

Table 2.2: Brain wave categorisation, frequency ranges and general attribution [78]

Brain Waves	Generally Attributed To	Frequency Ranges (Hz)
δ	Sleep	1-4
θ	Creative State or Drowsiness	4-8
α	Relaxed Focus or Daydreaming	8-13
β	Active Thinking or Problem Solving	13-30
γ	Excited or Irritability	30-100

Theta rhythms are typically found during problem–solving activities and are located in the pre-frontal cortex. Beta rhythms can be observed across the cortex. They are high–frequency and low–amplitude in nature and are commonly observed when the subject is awake. They

are often associated with busy, active states or active concentration. There are two type of beta waves - beta Rolandic rhythms and beta frontal rhythms. Beta Rolandic rhythms are associated with spontaneous activity and is generally observed around the sensorimotor region, where the $C3$, Cz and $C4$ electrodes are found. These waves are also related with intentionality of performing a movement, and presents desynchronisation before the mu rhythm. Beta frontal rhythms are those observed in the frontal lobe around the $F3$, Fz and $F4$ electrodes. In particular, they appear during cognitive tasks which relate to decision making For the purposes of the experiments conducted, beta Rolandic and frontal rhythms are classified as one.

Alpha waves were the first to be discovered. They are present in various locations in the cortex. However, they are particularly prominent in the occipital regions, where they are generally of higher amplitude than the other rhythms. The three categories of alpha rhythms found in the brain are mu rhythms, alpha occipital rhythms, and alpha parietal rhythms.

Mu rhythms have been widely implemented in BCI systems. Their namesake is due to the signals sharp negative peaks, much like the letter mu. Mu rhythms are found on both hemispheres of the brain, and sometimes appear as though they are symmetrical on a power map.

Alpha occipital rhythms are high amplitude signals recorded from the brain's occipital region. These are from the $O1$ and $O2$ electrodes. These rhythms have enhanced amplitudes when the subject's eyes are closed, and they decrease when in response to visual stimuli. Alpha parietal rhythms are enhanced when the subjects eyes are closed, much like the Alpha occipital rhythms. However this does not occur in all subjects – some were observed to have decreased amplitude when their eyes were closed. The Pz electrode is where this signal is most observable. However, the functional properties of the parietal rhythm are not fully understood. Again, much like the beta waves, for the purposes of the experiments conducted during the course of this research, alpha parietal and alpha occipital rhythms are classified as one, as seen widely in literature.

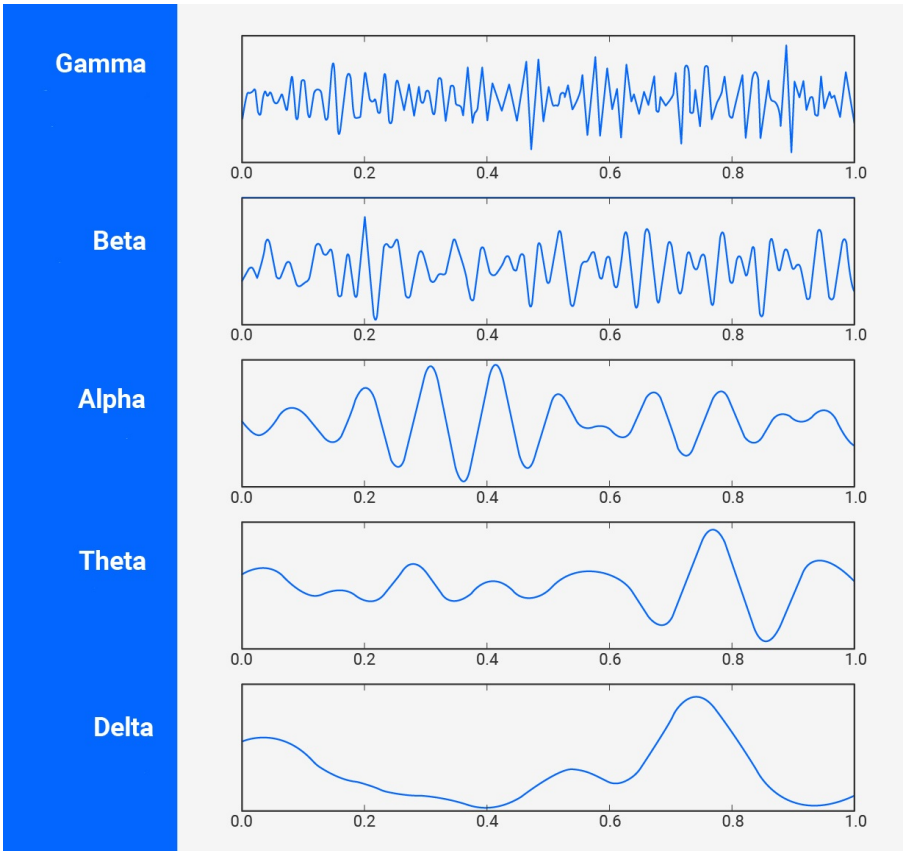


Figure 2.4: Types of EEG rhythms

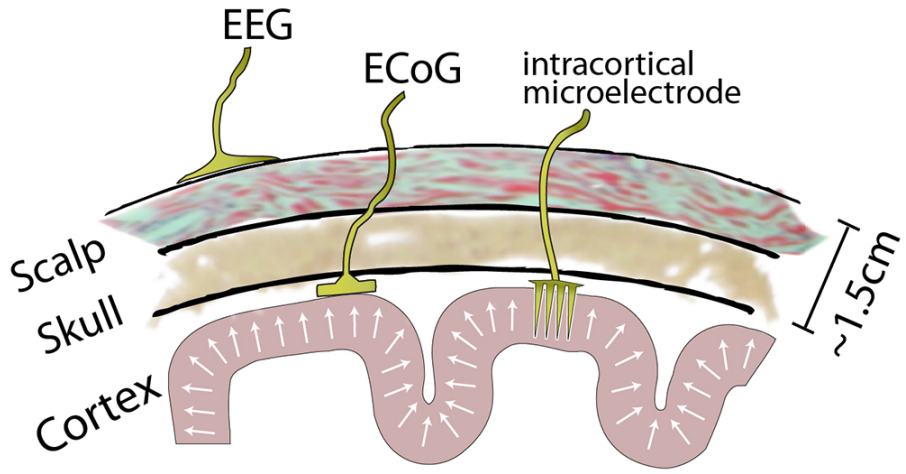


Figure 2.5: Electrode types that have been used for BCIs: EEG from scalp, ECoG from brain surface, and cortex-penetrating microelectrodes [79]

Figure 2.5 shows different invasive and non–invasive electrode types, and their placement during signal collection. EEG sensors are place on the surface of the skull i.e. non–invasive, and they are consequently the most commonly used electrode type in BCI research.

Invasive forms of electrodes include, Cortical surface electrodes and intracortical electodes. Non-invasive types of electrodes include fNIR, EEG, MEG and fMRI.

German physiologist Hans Berger recorded the first human EEG in 1924. He found that using radio equipment from his laboratory. He was able to amplify EEG signals detected on the scalp [77]. EEG is a method of recording the electrical activity of the brain.

It is a non–invasive procedure in which electrodes are placed on particular areas of the scalp. It measures the voltage fluctuations which occur from the resulting ionic current within the neurons of the brain [80]. The waveforms of the EEG reflect the post–synaptic potentials by cortical neurons in the brain.

It has been observed that different EEG signals are generated during the execution of varying tasks such as problem–solving [81], visualisation [55], and MI [82]. Each electrode records raw EEG data as a one–dimensional vector. The signals are obtained through volume conduction across numerous brain tissues on the three–dimensional scalp surface [83].

EEG has several advantages over other brain imaging approaches (shown in Table 2.3), despite its inherent disadvantages. Because of its non-invasive nature, EEG is low cost, portable, and generates no adverse effects [84, 85]. As a result, EEG offers a wide range of applications, including screening and hypothesis-based diagnoses. Several types of brain illnesses, such as epilepsy [6], tumours [86], Alzheimer's [87], sleep problem [88], and others, can be examined depending on the shape of waves, such as rapid spikes waves or slow waves.

Table 2.3: Properties of various signal acquisition systems

Imaging Technique	Direct/Indirect Measurement	Risk	Portability
ECoG	Direct	Invasive	Portable
EEG	Direct	Non-Invasive	Portable
MeA	Direct	Highly invasive	Portable
MEG	Indirect	Non-Invasive	Non-Portable
fMRI	Indirect	Non-Invasive	Non-Portable
NIRS	Indirect	Non-Invasive	Portable

In the medical field, EEG is used as a method of diagnosis of various brain-related conditions. Some possible diagnoses using EEG: Epilepsy, sleep disorders, depth of coma, and brain death. There are several advantages with the use of EEG relevant to this work. Firstly, EEGs are non–invasive which is highly desirable for recurrent testing and movement-based research applications where risk is higher. Additionally, EEG hardware costs are significantly lower to other techniques used to detect brain activity. EEGs also have a high temporal resolution, in order of milliseconds rather than seconds, which proves to be essential in terms of movement detection. EEG sampling rates are between 250 to 2000 Hz. It has additionally been observed that EEG devices are tolerant to subject movement, which is not the case for most other techniques. EEGs also have a lower magnetic field which enables them to be used for patients with metal implants and pacemakers.

However EEG devices also have several disadvantages, which are mainly related to their

non-invasive nature. Firstly, due to low spatial resolution it can be difficult to interpret which areas are activated by a stimuli. EEGs are unable to measure the electrical activity of the lower layers of the brain well. Additionally it can be tedious to connect a subject to EEG, and it would require the precise placement of many electrodes around the head, using various solutions to keep them in place. It is less taxing to prepare someone for other non–invasive recording techniques as a general rule. Finally, the Signal–to–Noise ratio is low, requiring complex data analysis to extract useful information from the EEG data. The EEG measures the voltage fluctuations caused by the firing of these electrical impulses [89]. These are then processed to find their spectral content. Extensive data cleaning and analysis requires to be performed to extract relevant information [90].

Traditionally EEG devices have been used mainly in the medical field, such as seizure activity [91], and monitoring patients post-surgery [86]. However, in recent years there has been much advancement in EEG technology, making it much more accessible outside of solely medical applications. Research is found in the area of social interaction [92], cognitive studies [93], neuromarketing [94] and BCI applications [95].

There are a few EEG consumer products available on the market. Some of the most popular are Emotiv [96] and Neurosky [97] headsets, as well as OpenBCI, a biosensing microcontroller used to acquire biological signals, the headsets blueprints are available for 3D print technology from online schematics.

Wet electrodes are the most often used clinical electrodes in EEG headsets. They employ a saline or gel environment to improve contact area, reduce impedance, and capture high-quality data [98]. Dry electrodes, on the other hand, are more suited to sustained signal capture. Dry electrodes have the fundamental advantage of requiring no prior preparation. These are primarily hands-on procedures that require personnel with EEG experience. Another significant discomfort is the annoyance experienced by the subject being tested. For example, despite being slightly invasive and barely hazardous, abrasive paste and electrolyte gel are sticky items that clog the hair and scalp [99].

Before driving any cabling, active electrodes amplify and buffer V-level EEG signals locally. Due to the AE's low output impedance, cable motion artefacts are minimised, allowing for high-impedance dry electrodes for increased user comfort [100]. Wet electrodes are regarded as the gold standard, and new dry electrodes must be evaluated against them before being claimed to be adequate.

During the earlier years of research with EEG, there were varied approached to electrode placement, however this lacked a uniformity when data was shared between research centres. A committee from the International Federation of Societies for EEG and Clinical Neurophysiology studied a variety of approaches, and in 1958 the 10–20 electrode placement system became internationally adopted.

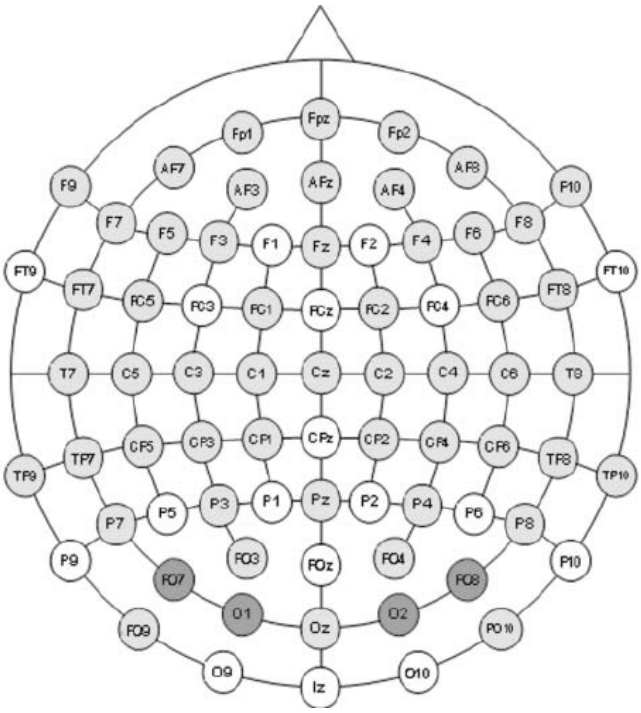


Figure 2.6: 10-20 Electrode Placement System [101]

The 10 and 20 refer to the percentage rather than absolute distance which allows for variance in head shape and size. There are other variants such as the higher resolution 5–10 system. The placement method of the electrodes in the 10–20 system can be seen below in Figure 2.6. Each of the letters F, T, C, P and O are used to identify the site location (Frontal, Temporal,

Central, Parietal and Occipital, respectively).

Electrocorticography (ECoG) was invented in the 1950s by neurosurgeons Wilder Penfield and Herber Jasper. The signals are recorded directly from the surgically exposed cerebral cortex without penetrating the blood–brain barrier. Initially, this was developed as a treatment for epilepsy. ECoG was also used to map and explore the brain's functional anatomy, such as identifying somatosensory and somatomotor cortex areas, and such that these sections would be avoided during surgery.

ECoG offers greater precision than EEG recording from the scalp due to closer proximity to neural activity and higher spatial resolution. However, there are several limitations of ECoG. Data sampling for ECoG is limited to the area exposed during the surgery, thus giving a limited field of view. Most significantly, the patient is under the influence of anaesthetic narcotic analgesics. This altered state has to be taken into account for the duration of the recording. Finally, the sampling time of ECoG is limited due to the invasive nature of the surgery.

2.5 Introduction to Traditional Machine Learning

ML can be segregated into five main tasks: classification, regression, ranking, clustering, and dimensionality reduction. Dependent on how a model is trained, ML can be also divided into supervised, unsupervised and semi–supervised learning. Supervised learning is when a ML model is given the labels of the dataset during training. In contrast, unsupervised learning is where a model is not provided with labelled data, and it is dependent on the model to find patterns within the data. Semi–supervised learning falls in between the two above learning methods. In this method the training sample has both labelled and unlabelled data [102].

Classification is an application of supervised learning, and it is the task of assigning a category to each item. In contrast, regression is tasked with predicting a value based on the data

it receives, such as the prediction of stock values. Regression also comes under the category of supervised learning. The last task considered under the classification umbrella is ranking, which is where a model is tasked with ordering items based on a given criteria.

Both clustering and dimensionality reduction are unsupervised training methods. Clustering is the partitioning into similar data groups, and is often used to analyse large datasets. Dimensionality reduction is where initial representations of data are transformed into a lower–dimensional representations, whilst still maintaining some properties of the initial representations of data. Some popular methods of traditional ML include Support Vector Machines, Random Forest, Logistic Regression, Decision Trees and ANNs.

Traditional ML requires pre–processing of the raw data and manual extraction of features from the data. Generally, following this process, the data is divided into training, validation and test sets. The training data is used to train the model, and validation data is used to monitor the performance of the model during training. Finally, the test set is used to check the model's generalisation performance. The pre–trained model can be fed with data and would have the ability to make predictions based on the patterns learned through the model training process.

2.6 Traditional Machine Learning for MI BCI applications

This section explores how traditional ML has been used to interpret MI EEG signals for applications within BCI. There have been several techniques which have been developed for feature extraction such as FFT, WT, standard deviation, variance, and power spectral density (PSD).

Conventional machine learning techniques have been successfully used to identify patterns in the collected EEG signal that correspond to someone is desire to move a limb or focus their attention on a flickering light.

BCI systems based on ML have been successfully used to control prosthetic arms [13], wheelchairs [103], and computer programmes. As with other biomedical data, EEG signals contain tens of thousands of features that can be analysed and interpreted to yield detailed information about various physiological functions [71].

Several different techniques have been used to extract EEG features over the last few decades, including standard deviation (SD), variance, FFT [104], WT [105], PSD [36]. Prior knowledge of neurophysiology is critical for identifying significant traits that will provide greater discriminative information between EEG classes [106].

Linear Discriminant Analysis, K-Nearest Neighbour (KNN), Artificial Neural Network (ANN), and Naïve Bayes are some of the most often utilised traditional machine learning models for EEG signal interpretation in BCI applications [107].

Wang et al. employed Fisher Linear Discriminant and Support Vector Machine (SVM) to understand and control a humanoid robot using MI-based EEG signals [108]. ANN and SVM were employed again to interpret EEGs caused by four different flickering frequencies presented on a LCD display for BCI applications, with SVM achieving an accuracy of 88.5% [109]. The performance of a machine in identifying patterns is highly dependent on the training data and the model itself [110].

2.7 Limitations of conventional EEG data analysis

This section highlights the shortcomings of traditional ML in the development of automated biomedical data analysis. The primary drawback of traditional ML is the requirement for human feature extraction. This section addresses the effects of manual feature extraction on the application of standard machine learning to EEG data and biological pictures.

Although there are several real-world applications of computerised EEG analysis based on traditional ML, the technique has always encountered certain challenges for the real life ap-

plications [111]. EEG data are known to be diverse, temporally dependent, sparse, irregular, noisy, and of high dimension, making feature extraction for training traditional ML models prohibitively difficult [112].

Determining the best representative feature set is a significant issue since it requires someone with extensive knowledge and expertise in the field of feature extraction. Additionally, the work of extracting strong and accurate features is always harmed by noise and interference from other signals, which is a typical source of limitation in EEG data.

The feature extraction procedure in EEG data is always influenced by variables such as motion artefacts [113], EOG, and EMG. EOG is a tiny electrical signal produced by eye movement, whereas EMG is a muscle electrical signal. For instance, in MI categorisation tasks, the imagined movement is frequently lost in the mix of other signals such as EOG and EMG [114].

Although several techniques for reducing feature dimensions have been applied, such as Principle Component Analysis (PCA), linking the features from the original feature space to the new features is a difficult task [115], as PCA assumes that the principle components are a linear combination of the original features [116]. The effectiveness of machine learning models trained on extracted features from collected EEG signals is constantly impacted by the extraction of redundant features [15].

2.8 Deep Learning

DL is an area of ML algorithms referring to the system of deep NNs, and is considered to be one of the pillars of AI [117]. To address the limits of conventional ML in order to produce more generalisation-oriented solutions. DL aims to replicate the human brain's operation by creating highly interconnected NNs and therefore producing a generic model capable of handling various input sources. DL is a fully integrated methodology, meaning raw data can be immediately input into the DNN for parameter and hyperparameters learning [118].

In other words, DL enables the direct feeding of DNNs with raw data and with little or no preprocessing; additionally, DL integrates feature extraction, selection, and classification into a single pipeline. Additionally, the DL approach is ubiquitous, robust, general, and scalable [119].

DL approaches have enabled researchers to classify, detect, recognise, and, most significantly, comprehend tasks that humans traditionally perform. Without the computing power of today's GPUs and cloud computing it would not have been possible to deploy DL algorithms, although they have provided a performance boost. Numerous new NN classes have been developed in image classification (most notably CNNs) and prediction analysis (RNNs). Due to the widespread availability of information over the internet, datasets can now be made open-source and become widely available.

DL approaches are dynamic and can be sensitive to changes in the data they analyse by considering the underlying patterns they discover. Similarly to how humans learn via experience, deep learning computers acquire knowledge by repeatedly executing a task and changing the algorithm to influence the outcome. To uncover all of the valuable information included in large datasets, in-depth analysis is required. Traditional classification techniques perform poorly in large and dynamic datasets due to their inability to capture all possible diversity states within the data [120]. DL strives to overcome the limitations of existing NNs by covering all information contained within a training dataset. However, training DNN requires many parameters and hyperparameters, which significantly increases training time compared to other approaches and consumes prohibitively more hardware resources [121, 122, 123]. Nonetheless, it is possible to address such issues by using devices with high computing power.

Numerous investigations established DL's superiority to alternative state-of-the-art techniques. In [124], researchers used a short-term Fourier transform (STFT) to translate MI EEG signals into two-dimensional time-frequency spectrum images for training a CNN. The classification findings indicated that CNN outperformed SVM and ANN classifiers. They

utilised a CNN to extract spatial features from MI EEG data and then an LSTM to extract temporal features from signals in [124]. This framework outperformed the SVM classifier. In [125], they used a CNN to extract temporal and spatial features from MI EEG data and then auto encoders to fuse the features; this study demonstrated that this framework performed better than one that used FBCSP as the feature extractor and three different classifiers, namely Naïve Bayes, Linear Discriminant Analysis, and SVM.

The following subsections discuss DNNs in detail, including the various architecture types, input formulation kinds, learning methodologies and deep learning platforms. Common optimisation methodologies are detailed in C.6.

2.8.1 Convolutional Neural Networks

Convolutional Neural Networks are more commonly known as ConvNets or CNNs, and were inspired by the visual cortex. These are deep feed-forward NNs, made with a complex combination of NNs. They are particularly adept at finding patterns within images and classifying them. However, the drawback of this approach is that CNNs are not designed to capture the underlying relationships between different parts of the input data that are not local in nature. In particular, if the data has a high degree of permutation invariance, i.e. if the order of the elements in the data does not significantly impact its semantic meaning, then the local spatial patterns captured by a CNN may not be as informative as more global relationships between the elements.

CNNs are capable of robustly recognising the pattern and structure of image and time series. Aside from being effectively employed in computer vision and natural language processing, CNN structures have also shown tremendous promise in biomedical signal processing. CNN had not gained much attention before 2012 because of data and technical restrictions. However, the discovery of "AlexNet" [126], which has addressed a wide range of applications with surprisingly strong results, has prompted researchers to investigate several DNN models, including VGG [127], GoogleNet [128], and ResNet [129].

A CNN model required three stacked layers: a convolutional layer, a pooling layer, and a fully connected layer. Convolutional layer convolves the tensor to the shape, while the pooling layer reduces the necessary calculations to lower the data dimension. Finally, the completely linked layer connects each neuron from the previous layer to a new layer that will be used to forecast or classify the objective goals. When talking in terms of the use of DL and EEG, this approach of applying DL with stacked layers aims to reduce the amount of input data and capture the EEG patterns' distinctive spatial dependence.

Significant advancements in CNN classification tasks are critical for the development of the BCI, including enhancements to EEG-based fatigue evaluation. For instance, a spatial-temporal CNN system based on EEG was proposed in [130]. This system's advancement extracts temporal dependencies. It is used in conjunction with dense layers to process and classify spatial-temporal EEG data, achieving a classification accuracy of 97% or EEG driving tiredness levels. Another recent study [131] used a visual EEG experiment to detect various levels of weariness. It suggested a multi-scale CNN system that extracts raw EEG signals from an inception structure using a spacetime frequency combination features extraction technique.

Generally, it is not commonplace to supply a DL model with raw EEG data, as varied noise distributions among the signals makes it difficult to extract straightforward features from the original EEG signal. Since EEG signals are time–based, they also require to be transformed appropriately to be compatible with CNN models.

Within the literature, there is not a singular preferred methodology for model creation. Dose et al. [132], proposed a method for a shallow network where the training samples were first applied on the time domain, and then applied a spatial filter along the EEG channel dimension which would created a weighted linear dimension of a single channel, followed by a pooling and flatten layer, leading to a classification layer. Their method presents good results – with a mean cross–validation accuracy of 80.10%. It was discussed that DL architectures would be a viable alternative in order to find the rather more intricate features.

The need for preprocessing becomes clear, as raw EEG signals are hard to decrypt on their own. As a result, researchers such as [133] additionally pre–processed the data and filtered the signals using a Butterworth band–pass filter, and a zero phase digital filter in order to obtain signals of various brain signal frequency bands (alpha, beta, gamma, theta, delta). The model implemented was also of a shallow network. Though pre–processing was applied here, the results shown were less than ideal, with the average result at 61.3%. They compared the results classic SVM algorithms, which showed poorer results at 51.5%.

The use of Fourier transforms, in order to bring the EEG signals into the frequency–domain, has become an increasingly popular method of feature extraction. Wang et al, [134], extracted three electrode data points in the central region of the motor–cortex (not–specified), apply a STFT. These were then transformed into images of on the spectral data on time stamps. Here however all frequency points between $8 - 30Hz$ were taken into account, and visualised as what looks like a 'pixelated image' [134]. The deep network architecture consisted of 7 layers, two sets of convolutional layers + max pooling, followed by two dense layers used to compute the predicted labels, the average accuracy was at 86.74%. They compared this value to vanilla CSP + SVM, to which is outperformed with a mean accuracy of 9.05%.

2.8.2 Recurrent Neural Networks

The number of studies investigating the RNN or LSTM networks when applied to EEG-based BCIs has expanded significantly in recent years, with data indicating that RNN or LSTM-based approaches outperform classical ML. The reason for this is that they are capable of extracting hidden EEG feature presentations, and when paired with CNN, they may optimise feature representations by including both temporal and spatial information included in EEG signals. The RNN framework is beneficial for identifying motion, recognising emotions, and assessing sleep stage. For instance, Supratak et al. [135] examined automatic sleep stage classification using a bidirectional LSTM algorithm for learning tran-

sitional rules from hidden time series EEG features and a CNN for extracting time invariant EEG features.

RNNs have been designed mainly to handle sequence prediction problems. It is the first algorithm which has been able to remember the input due to internal memory, enabling them to forecast future events. The recurrent (lateral) connections of RNNs allow them to learn broader abstractions from input sequences. RNN's potential was only recently exploited more since the increase in available computational power.

In general, RNNs have two major obstacles which they have or have had to deal with: Exploding Gradient and Vanishing Gradient. Exploding Gradients occur when the algorithm randomly assigns high importance to weights without much reason, however this may be easily solved by truncating or squashing the gradients. Vanishing gradients occur when the gradient is so small that the model stops learning or takes a very long time to learn. The LSTM was the solution to this, and was first introduced by Sepp Hochreiter and Juergen Schmidhuber [136].

LSTM are an extension to RNNs - they have the additional behaviour where they are able to handle long-term dependencies. This is due to LSTMs having a computer–like memory which enables them to read, write and delete information from their memory. As such, LSTMs are well suited to learning more important information which may have long periods of lag between them. The memory of an LSTM can be seen as a gated cell. Here, gated is in reference to the cell deciding whether to store or delete information based on the importance given to it, which is controlled through weights. The problem of vanishing gradients in RNNs, is solved by LSTMs, because the gradients are kept steep enough, and therefore the training is relatively short, with higher levels of accuracy.

2.8.3 Hybrid Structures

Hybrid NN architectures refer to the combination of multiple different neural network architectures into a single model. These hybrid models can leverage the strengths of different NN architectures to achieve improved performance over using a single architecture alone.

In the context of EEG-based BCI classification, hybrid NN architectures have the potential to address some of the challenges faced by single architecture models. EEG signals are non-stationary and highly variable, which makes their analysis challenging. Additionally, EEG signals are typically measured in multiple channels, which requires the use of spatial filtering techniques to extract relevant features from the signals.

Hybrid NN architectures can help address these challenges by incorporating different neural network architectures to capture different aspects of the EEG signals. For example, a hybrid model could incorporate a CNN for spatial filtering, followed by a RNN for temporal analysis. The CNN would be used to extract relevant features from the EEG signals, while the RNN would be used to model the temporal dependencies in the signals.

Hybrid architectures for biological signal analysis has recently been of interest for researchers to explore if combining architectures improves results of model implementations. Due to these developments in hybrid NN being quite recent, the currently available literature is sparse, in particular, in relation to EEG hybrid architectures.

In [137], Wang et al. create power spectral images from the raw EEG data, and transform these into images which look much like heat maps of the brain. The network architecture is made up of a combination of CNN and LSTM networks, where the CNN is used to extract the space and frequency features and the LSTM network is used to extract the timing features of the EEG signal. The authors present the advantages of their technique as making use of the features which are prominent in both the frequency and time domains.

An architecture proposed by Hu et al. [138], suggested a 11 layer CNN structure, followed by an LSTM layer, and finally by fully connected layers. They compared the accuracy of

their models to state of the art DL representations and found that the model implemented was 8.9% better in terms of accuracy, in comparison to a normal non–hybrid network implementation, and 11.4% higher than traditional ML approaches.

Bresch et al. [88], proposed a novel approach using DNNs to classify different sleep stages, which is normally visually classified by a human annotator. Their architecture consisted of a deep network; a structural idea which was also used by Hu et al. [138]. Their findings suggest that the accuracy of their hybrid network was comparable to traditional DNNs. Though it was also found that, for the most part, the DNN and human annotators agreed on the classification at most stages, this was not always the case.

2.9 Preprocessing

Preprocessing techniques used in BCI focus mostly on the frequency domain and spatial filtering. Band-pass and notch filters are the most often used methods in frequency domain filtering because they can extract the distinctive signals situated at the stimulus frequency and eliminate noise and artefacts. These filters are constructed per the frequency characteristics of the associated signals. The band-pass filter's frequency range is often determined by the stimulation frequencies or their harmonics, whereas a notch filter is utilised to eliminate power line interference.

Processing numerous channels of brain signal data, spatial filters can increase the signal-to-noise ratio of the brain signal response. Additionally, spatial filtering can be utilised to extract features. Raw EEG data normally have a lot of unwanted background noise that needs to be removed before the meaningful analysis can begin. There are various pre-processing steps which are seen throughout literature [139, 140, 141].

Filtering is the first common method to pre-process signals. One possible method is using a notch filter to remove power line noise at 50Hz or 60Hz. This is dependent on the country where the signal recording has taken place. The second method is using a high-pass filter

with a low cut-off frequency in order to remove the baseline drift. The third common filtering method is using band-pass filters to select specific frequency bands. Another common pre-processing step is to clip the amplitude of EEG signals by forcing them to fall within a certain range or using the mean and standard deviation. A similar method is to cancel some of the samples from the start and/or finish of the signal to eliminate potential acute artefacts.

Other pre-processing approaches include using artefact rejection techniques for noise created by electroocoulogram and electromyogram artefacts. This popular method of EEG pre-processing works by referencing signals using either an electrode or by the average of signals from all the electrodes, and normalising the data to a zero mean and unit variance using the z-score. This technique potentially can avoid a local minima trap and could speed up convergence. Finally, by reducing the overall number of electrodes from which the data is collected you could reduce the overall amount of noise.

EEG signals are transmitted through bone, tissue, hair and other barriers before they make contact with the electrodes and can be recorded. An artefact is denoted as a component of the EEG signal which is not produced directly by human brain activity [142]. Therefore an artefact can be defined as noise generated by the human system, which alters the neural EEG data [143]. Physiological artefacts originate from the subject, and non-physiological artefacts originate from the environment around the subject. [144]

Table 2.4 lists possible physiological and non-physiological artefacts which may occur during the recording of EEG signals. Each artefact mentioned in Table 2.4 is discussed with detail in Appendix A.2.

Table 2.4: List of Physiological and Non-physiological artefacts

Physiological artefacts	Non-physiological artefacts
Ocular activity	Electrode pop
Muscle activity	Cable movement
Cardiac activity	Incorrect reference placement
Perspiration	AC electrical and electromagnetic interferences
Respiration	Body movements

2.10 Feature Extraction Methods

The extraction of features is a critical challenge in signal processing and is critical to the whole BCI system. Numerous approaches have been employed in various EEG paradigms. The following sections cover many frequently used feature extraction methods which includes Fourier-based transform, WT, Hilbert-Huand transform, Independent Component Analysis (ICA) and CSP.

The Fourier Transform is comprised of the FFT and the DFT. Fourier Transform techniques are mostly used to analyse the PSD. FFT is a fast DFT computation algorithm, which may have an effect on practical applications. In real-world applications, the available stimulation frequencies may be limited due to the frequency resolutions being constrained by the length of the data segment.

Among the benefits of employing FFTs are simplicity and a short computation time. Wang et al. employed 256-point FFT to translate EEG signals into the frequency domain representing five frequencies in SSVEP-based BCI [145]. When respondents were not focused on any stimuli, a 128-point FFT averaged the three spectral components around the target frequency. The average value was used to determine whether a subject was in an idle condition. Mouli et al. used the FFT's maximum amplitudes to differentiate between multiple target stimuli at frequencies of 7, 8, 9, and 10 Hz [146]. Muller-Putz and Pfurtscheller esti-

mated the frequency components of the EEG signal using split-radix FFT and then averaged the three spectral components around the target frequency [147]. Additionally, Diez et al. employed the FFT to estimate the power spectral density derived from the DFT [148]. All of these studies that used FFT to estimate DFT demonstrate FFT's computing advantages. FFT is widely used in SSVEP systems throughout a wide frequency range, from low to high. Due to its short calculation time, FFT is frequently used to estimate DFT.

EEG signals are non-stationary in nature, with frequency components that fluctuate with time [149]. Wavelet Transforms, which allow adjustable time-frequency resolution, can aid in the analysis of such data. WT is a Fourier analysis with adjustable-windows that is based on FT [150]. The advantage of WT over FT is the ease with which different mother wavelet functions may be used to evaluate various signal types. Due to its capacity to adapt to signal components and its multi resolution, WT is widely used to analyse EEG signals. Zhang et al. incorporated the Continuous Wavelet Transform (CWT) into SSVEP feature extraction and classification in SSVEP-based BCI [151].

The choice of the mother wavelet is critical in CWT. They compared the performance of various wavelet types in SSVEP classification. Experiments revealed that the Complex Morlet wavelet outperformed others and excelled at segmenting brief EEG data. Kumari and Somani determined the feature vectors using the coefficients of CWT to find the location of high frequency components in SSVEP [152]. The CWT provides a representation of EEG signals in the time-scale domain in MI-based BCI [149], and thus can be used to precisely localise ERD/ERS components in the time-scale domain. Hsu and Sun used CWT in conjunction with Student's two-sample t-statistics to extract 2D time-scale features [153].

HHT is a recently created adaptive data analysis method that combines empirical mode decomposition and Hilbert spectral analysis [154]. It has been extensively used in EEG research. Unlike FFT, which is based on cosine functions, HHT is self-adaptive and can improve performance in certain signal segments, allowing it to be utilised for both stationary and non–stationary signals analysis. However, the computation time for HHT is longer than

for FT.

ICA is a relatively new technique that aims to establish a linear representation for non-Gaussian data in order to ensure that its components are statistically independent or as independent as possible. In a variety of applications, such as feature extraction and signal separation, this format appears to capture the data's essential structure [155]. ICA is frequently used in preprocessing to extract noise/interference from raw EEG signals. ICA, on the other hand, typically integrates additional feature extraction techniques in order to classify the various targets in distinct EEG scenarios.

CSP is a powerful signal processing technique that has been proven to extract more discriminative information than bipolar, Laplacian, or CAR spatial filters [156]. The CSP principle results in a set of spatial filters that are optimised for one class while minimising variance for another. CSP can effectively suppress noise by utilising data from a large number of electrodes and so requires a small number of electrodes to perform well. CSP is a powerful technique, particularly for MI classification. Numerous new CSP-based approaches have been proposed in recent years to improve classification accuracy. Samek et al. [157] devised a technique termed stationary CSP that regularises the CSP solution towards stationary subspaces; in other words, the CSP is extended to be invariant to data non-stationarity.

The advantage of using deep learning is the ability to learn EEG signals end-to-end, extract their features, and classify them automatically, thus making manual feature extraction, normally required for traditional ML techniques, redundant. However, given the incredibly complex nature of EEG signals, incorporating some manual feature extraction techniques into the pre-processing structure prior to passing these signals to the DNN can be a very powerful tool.

CNNs, RNNs, and hybrid network designs are all employed for EEG recognition. Not only can deep learning-based EEG signal identification learn features manually retrieved, but it can also learn features automatically from the original EEG signals or two-dimensional (2D) images translated from the frequency domain. As a result, deep learning technology

holds greater promise for the recognition and categorisation of EEG data than traditional ML-based methods.

2.11 Databases

This section discusses the open-source databases used in this research. The information obtained from understanding these datasets set the potential and limitations of the experiments. The dataset structure, experimental structure, and pre–processing methodologies will all be discussed during the course of this section.

Databases can serve as a basis for comparison after data has been collected. This is especially useful when initial data collection establishes a benchmark for evaluating patterns in signals. These datasets have not only been expertly gathered and thoroughly checked, but they have also been utilized by multiple BCI systems evaluations, making them the perfect choice for verifying the effectiveness of proposed MI event classification methods and comparing them to current techniques.

In particular, one database draws interest – the EEG database for BCI applications [158]. This included data sets such as four-class MI, P300 speller with ALS patients, and visual P300 speller. The currently available open–access datasets for MI based EEG are limited. The database which was used for the development of the models was a GigaScience Database by Ho et al.[159]. This was chosen mainly due to the number of participants available as part of one database and the full EEG cap electrodes of 64 channels. It is also the newest dataset available, where data was collected in 2017. Table 2.5 compilation of these databases.

The model implemented will be further tested against the BCI-IV datasets in Table 2.5, as seen in literature – these are the datasets that are typically used during implementation or as a testing benchmark for the models made. Producing the EEG signal is critical for a successful BCI in MI. As a result, the problem of human training requires consideration. In comparison to SSVEP and ERP, MI requires a longer training period to produce the ERD/ERS

phenomena. To enable a subject to take control of the EEG response, some form of feedback is required, particularly at the start. Typically, the MI training process includes both offline and online instruction. Offline training is required to fine-tune the user's EEG signals and train the recognition algorithm.

Table 2.5: Detailed open-source EEG datasets

Dataset	No. of Subjects	Sampling Rate (Hz)	No. of Electrodes	MI Classes
BCI III - Dataset IIIa	3	250	60	4 - Left/Right Hand, Foot, Tongue
BCI III - Dataset IIIb	3	125	2	2 - Left/Right Hand
BCI III - Dataset IVa	5	1000/ 100	118	2 - Right hand/Foot
BCI III - Dataset IVb	1	1000/ 100	118	2 - Left hand/Foot
BCI III - Dataset IVc	1	1000/ 100	118	2 - Left hand/Foot
BCI III - Dataset V	3	512	32	3 - Left/Right Hand, Word Image
BCI IV - Dataset I	7	1000	64	3 - Left/Right Hand (either), Feet, Tongue
BCI IV - Dataset IIa	9	250	22	4 - Left/Right Hand, Feet, Tongue
BCI IV - Dataset IIb	9	250	3	2 - Left/Right
GigaDB	52	512	64	2 - Left/Right Hand

2.11.1 BCI Competition III - Dataset IIIA

Graz University of Technology provided this dataset as part of the BCI Competition III [158]. The recording was made with a 64-channel Neuroscan EEG amplifier, with the left mastoid serving as a reference and the right mastoid serving as ground. The EEG was captured at 250 Hz and filtered with Notch filter between 1 and 50 Hz. Figure 2.7 shows the recording technique for sixty EEG channels, which are not a traditional, and internationally recognised mapping system.

These subjects were asked to perform an imaginary left or right index finger, as shown in Figure 2.8. The aim was to perform visualisations with the left and right hands while following a signal. The experiment consists of several runs with 40 trials each. Each of the four MI tasks were displayed 10 times, in a randomised order.

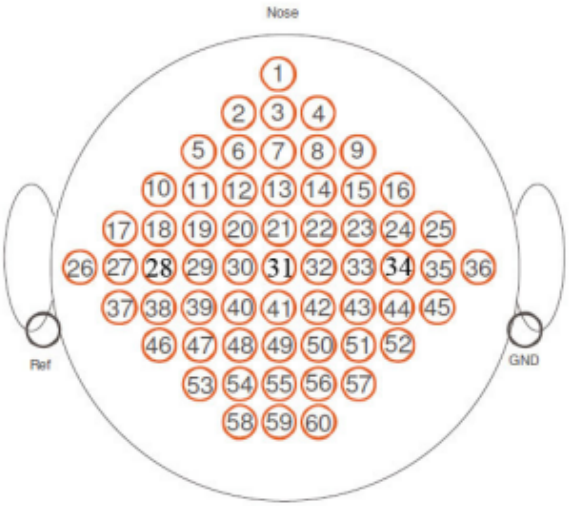


Figure 2.7: BCI III - Dataset IIIa 60 channel electrode positions [158]

Dataset IIIa includes three subjects: k3b, k6b, and l1b who were asked to perform four different MI tasks – repetitive movement of right hand, left hand, tongue or foot. The paradigm and time scheme were as follows. The first $2sec$ of the trial were silent. At $t = 2sec$ an acoustic stimulus indicated the start of the trial, and a cross "+" was displayed. From $t = 3sec$, an arrow to the left, right, and up was displayed for $1sec$. At the same time, the subject was asked to imagine the specified movement up until the cross disappeared when $t = 7sec$.

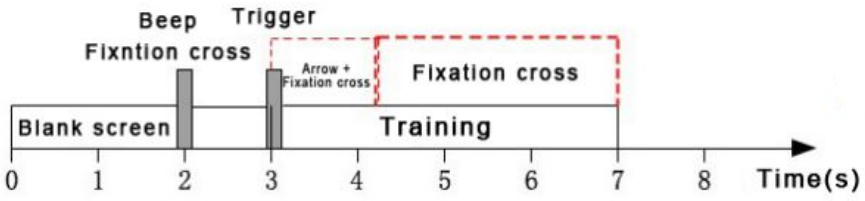


Figure 2.8: Timing paradigm of BCI III - Dataset IIIa [158]

2.11.2 BCI Competition III - Dataset IVA

This dataset was provided by the Frauhofer FIRST, Intelligent Data Analysts Group and Carite University Medicine Berlin [158]. The EEG signals were recorded on a 118 channel EEG. Signals were band pass filtered between 0.05 Hz and 200 Hz, and then digitised at

1000 Hz. The dataset is also available at a down-sampled rate of 100 Hz. This was the chosen dataset version for this research.

Five healthy subjects participated in this experiment. Each subject performed 2 different MI tasks: right hand and left foot Visual cues were indicated for 3.5 seconds which of the three MI tasks which The presentation of the target cues were randomised between 1.75 to 2.25 seconds between which the subject could relax. The participants had two types of stimulation. First where targets were indicated with letters appears behind a visual fixation cross. Second, where a randomly moving object indicates the target.

2.11.3 BCI Competition III - Dataset IVB

This dataset was provided by the Frauhofer FIRST, Intelligent Data Analysts Group and Carite University Medicine Berlin [158].The EEG signals were recorded on a 118 channel EEG. Signals were band pass filtered between 0.05 Hz and 200 Hz, and then digitised at 1000 Hz. The dataset is also available at a down-sampled rate of 100 Hz. This was the chosen dataset version for this research.

There was one subject per task for this experiment. Each subject performed a different MI task: left hand, right hand, tongue (relax). The same paradigm to that of BCI Competition III - Dataset IVA was used for this experiment.

2.11.4 BCI Competition III - Dataset V

This dataset was provided by the IDIAP Research Institute. The data was collected using a BioSemi system with 32 electrodes [158]. The sampling rate of the data was at $512Hz$.

Three healthy subjects took part in this MI data collection experiment for 4 non–feedback sessions. The subjects were asked for the duration of the experiment to have their arms

relaxed on their legs. The three tasks consisted of imagination of a repetitive right and left hand movement, and the generation of words beginning with the same random letter.

All four sessions were acquired on the same day. Each session lasted 4 minutes, with 5–10 minutes between each. The subject performed a given task for approximately 15 seconds and then switched randomly to another task at the request of the operator. During the course of the 4 minute session the subject was continuously performing an MI task.

2.11.5 BCI Competition IV - Dataset I

This dataset was provided by the Berlin BCI group affliated with the Berlin Institute of Technology. The EEG signals were recorded on a 59 channel EEG, most densely distributed over the sensorimotor area. Signals were band pass filtered between 0.05 Hz and 200 Hz, and then digitised at 1000 Hz. The dataset is also available at a down-sampled rate of 100 Hz.

A total of 7 healthy subjects performed two types of MI activity: 2 classes of left hand, right and foot. The EEG data collected during the MI task (3 seconds or 750 samples each trial) was utilised to classify the participants in this study. Trials of the left-hand and right-hand MI tasks were retrieved for the study from the four classes of MI.

Throughout the session, MI was done in the absence of feedback. Two MI classes were chosen for each subject from the three available: left hand, right hand, and foot. The individual was directed to do the cued MI task for four seconds while the cues were displayed. These periods were interspersed with two seconds of blank screen and two seconds with a fixation cross displayed in the screen's centre. However, for the purposes of this study, we used solely right and left hand MI data.

2.11.6 BCI Competition IV - Dataset IIa

This dataset was provided by Graz, Austria [160]. EEG signals were captured at a sampling rate of 250 Hz from 22 electrodes. Placement of the electrodes was using standard 10–20 system. The raw data was sampled at 250Hz and band–pass filtered between 0.5Hz–100Hz. An additional 50Hz notch filter was used to suppress mains noise.

There were a total of 288 MI task trials, with 72 trials for each MI class. Data was collected over the course of two sessions in two days, where a session is defined as six runs, with a short break between them. A list of trials which contained artefacts was provided. These trials were discarded during the pre–processing of the raw data and not used during analysis.

Nine healthy subjects performed four types of MI activity: left-hand, right-hand, foot, and tongue. The EEG data collected during the MI task (3 seconds or 750 samples each trial) was utilised to classify the participants in this study. Trials of the left-hand and right-hand MI tasks were retrieved for the study from the four classes of MI.

At the start of each session approximately 5 minutes of EEG baseline data was collected to understand the influence of EOG. The EOG data was collected by an additional 3 monopolar EOG At $t = 0s$ a fixation cross appeared on a screen. In addition to this a short acoustic warning presented to indicate the beginning of the session. After $t = 2s$ a cue in the form of an arrow pointing either left, right down or up. These corresponded to the foot and tongue, respectively. The cue appeared and stayed on screen for 1.25s. This prompts the subjects to perform the desired MI for recording. The subjects were expected to carry out the MI task until the fixation cross disappeared from the screen at $t = 6s$. A short break followed this.

2.11.7 BCI Competition IV - Dataset IIb

This dataset was provided by Graz, Austria [160]. The data is made up of three bipolar recordings (C3, CZ, and C4) that correlate to the image in Figure 2.9a. The data was sampled

at the rate of 250 Hz. The raw data was bandpass filtered between 0.5 and 100Hz and with a notch filter of 50 Hz.

Nine healthy subjects took part in 5 sessions of EEG data collection. In the first two sessions training was conducted without feedback, whereas the last three sessions training was recorded with feedback. Subjects were instructed to visualise left or right hand movement. Each session consisted of six runs with ten trials each and two MI tasks. Thus a total of 20 trials per run and 120 trials per session. Each session began ($t = 0s$) with a short acoustic tone. For the first two sessions, at $t = 3s$ a cue showing an arrow pointing either left or right was presented for 1.25 seconds. Subjects were expected to imagine the corresponding hand movement for 4s. A pause of up to 1.5s followed the MI task.

During the three online feedback sessions, a smiley was used as a feedback visual tool. The smiley would be shown from the beginning of the session. At $t = 2$ a warning acoustic tone would sound. The visual cue was presented from $t = 3s$ to $t = 7.5s$. The smiley would change to green when the MI task was performed correctly, otherwise would remain grey. At $t = 7.5s$ the screen would go blank, and a random interval of 1 to 2 seconds was added between the current trial to the next. Figure 2.9b depicts the time scheme paradigm used during acquisition.

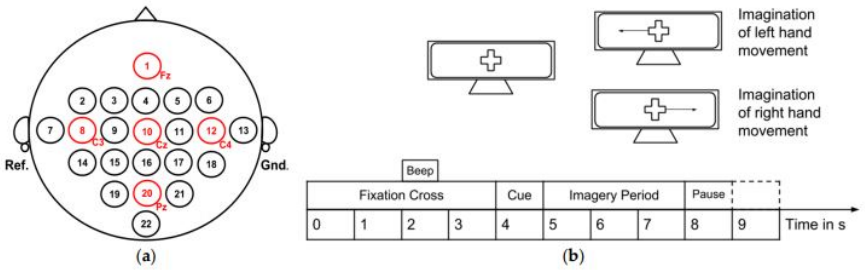


Figure 2.9: This figure includes information about the acquisition of dataset IIb from the BCI Competition IV, where (a) shows the electrode placement on the scalp (C3,Cz,C4) . and (b) shows the time scheme paradigm which was followed during data acquisition [160]

2.11.8 H.Cho Two-Class Dataset (GigaDB)

This dataset was collated by Hohyun Cho et al. [161]. The data was collected using the Biosmi ActiveTwo 64 channel EEG system, with a sampling rate of 512Hz. Two EMG electrode data points were recorded alongside EEG.

There were a total of 52 healthy participants. Before the MI experiment began, the subjects were asked to move their fingers starting from index finger and processing their little finger each touching their thumb, each after 3 seconds of onset. The subjects were asked, whilst during the MI ask, to imagine the kinaesthetic experience rather than imagining the visual experience.

The participants were asked to first perform real hand movement. The corresponding EEG data was collected during these actions. At the start of each trial, the monitor showed a blank screen with a fixation cross for 2s. The prompt for either left hand or right hand was then shown on screen for 3 seconds. A rest period, showing a blank screen, was shown at a random rate between 4.1 to 4.8 seconds. This process was repeated 20 times. Following the physical movement task, the MI experiment was conducted. Subjected were asked to imagine either the movement of the left or right hand. Either five or six runs were conducted during each MI experiment. Between each run a maximum of 4 minutes was given as a break period.

3

Deep learning using EEG spectrograms for classification of user intention

3.1 Introduction

In this chapter we explore a deep learning approach inspired by the recent successes in image classification using dCNNs designed to exploit invariance and capture compositional features in data [162, 163]. These systems have primarily been used to handle image data or audio data. This research aims to work with spectrograms of EEG data (2D time–frequency maps). This approach essentially treats the EEG channel data as an audio representation, and the proposed method mimics similar uses in deep networks of that area.

The algorithm was inspired by a paper which was classifying music into various categories using deep convolutional networks on spectrograms, [164]. This paper first applied windows onto music signals, created spectrograms from these, and then passed these images through a deep CNN architecture with 10 hidden layers. Their proposed network produced results of at least 90% accuracy which was on par with other state-of-the-art methodologies

in their research space.

Various noise signals degrade the original EEG during signal capture. As a result, pre-processing techniques aid in the removal of undesired artefacts from the collected EEG signal and increase the signal to noise ratio. The frequency ranges include the mu and beta bands, which have been demonstrated to be critical for MI categorisation.

Spectrograms display how the spectral density of a signal varies with time. The spectrogram representation of a one-dimensional signal captures the fluctuation in the signal's spectral content across time. The resulting two-dimensional image has time as the horizontal axis, frequency as the vertical axis, and the amplitude of the frequency components at a given moment as the intensity or colour of that point on the graph [165]. EEG signals are non–periodic signals and consequently need to be captured as consecutive snapshots using sliding windows.

The spectrogram was generated for the pre-processed EEG channels in their entirety. As noted previously, the spectrogram band-pass filtered the range. The reason for using an overall image rather than a stacked image as previously described was to create a more robust model that could generalise not just to a single participant or dataset, but to several datasets.

As mentioned in the literature (Chapter 2) there have been many proposed feature extraction methodologies for the classification of EEG MI signals. However, the majority of these techniques have limitations and are insensitive to EEG signal deformations. They overlook the class-relevant components contained in the time-frequency data. In other words, certain frequency components were excluded from the analysis because they did not contribute to class discrimination. This research mitigated these issues by utilising deep learning techniques for feature learning and classification.

The spectrogram is used to recover EEG data in the time-frequency domain by applying a STFT to the signal. By adjusting the temporal frame with some overlap, a process known as windowing, STFT is used to partition the EEG signal into several segments of short-

time signals [166]. A spectrogram is characterised as narrowband or wideband depending on the temporal windowing function w[n]. The Fourier transform is wideband when the time window is short, but the Fourier transform is narrowband when the time window is extended [167]. The STFT general equation of a signal S is given by equation 3.1.

$$S(m,k) = \sum_{n=0}^{N-1} s(n+mN)w(n)e^{-j(\frac{2\pi}{N})nk} \tag{3.1}$$

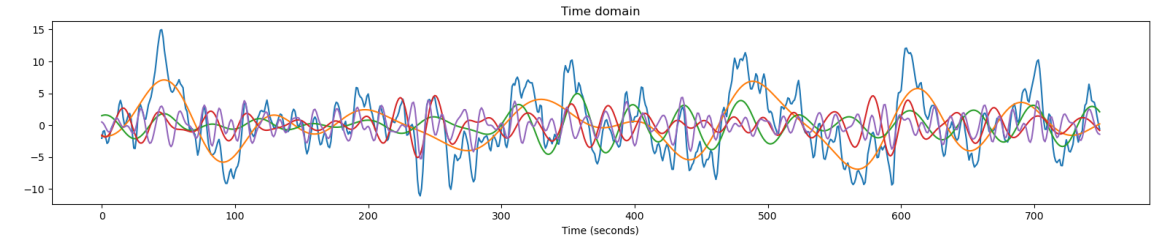
Where $k[0:K]$ is the kth Fourier coefficient. $K = N/2$ is the frequency index corresponding to the Nyquist frequency. $S(m,k)$ indicates the m-index time–frequency(frame) spectrogram. N = window segment length. mN = the shifting step of the time window. $w(n)$ = windowing method of an N-point sequence. mN should be at a value smaller than N to produce an overlap between time windows. S depends on the type of window function used such as Symmetric, Uni-modal and Gaussian.

A wide window gives less localisation in time but greater frequency discrimination in the spectrogram, which compromises time resolution and frequency resolution. The window captures a time slice of the signal with near–constant spectral properties [167]. The obtained segments then shift the time window whilst also overlapping. The spectrogram is defined as a magnitude of $S(m,k)$; this is represented at $A(m,k)$, shown in equation:

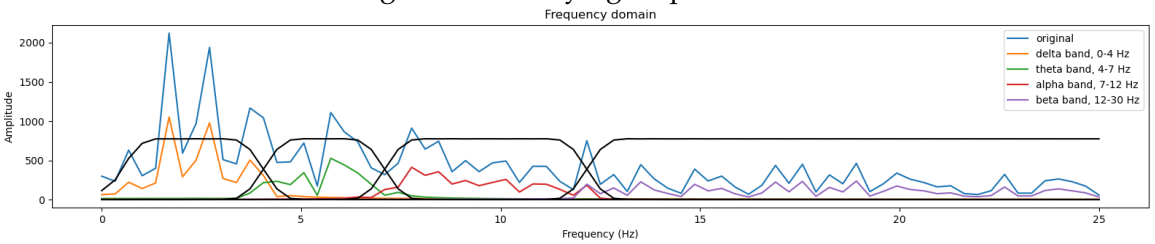
$$A(m,k) = \frac{1}{N}|S(m,k)|^2 \tag{3.2}$$

The resolution of the spectrogram can be improved by changing the length window; a longer window provides better frequency resolution but poor temporal resolution. A small window, on the other hand, offers better temporal resolution but low-frequency resolution. The selection of an adequate window length and overlapping is critical for a decent spectrogram visualisation. A spectrogram of a signal, a time–varying spectral representation of a signal, is shown in Figure 3.1. The following is a typical spectrogram layout: The x–axis represents time, the y–axis shows frequency, and the third dimension, colour-coded, represents

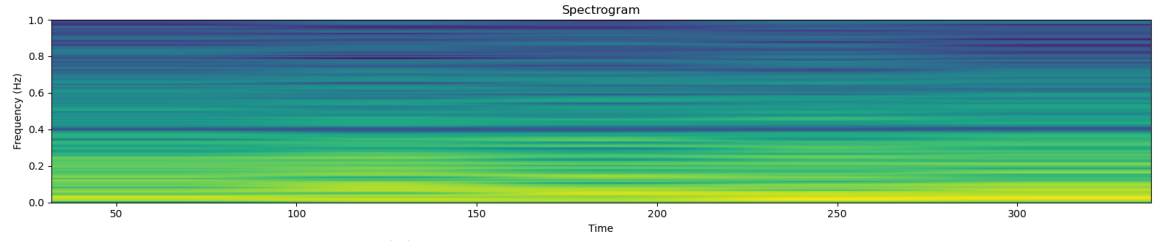
the amplitude (spectral content) of a frequency–time pair. Figure 3.1a shows overlapped EEG signals in the time domain, over the course of a particular period of time. The frequency domain representation of a signal refers to the representation of a signal in terms of its frequency components (Figure 3.1b), typically obtained through a Fourier transform. This representation, taking the form of a frequency versus magnitude or power graph, shows the magnitude and phase of the signal at each frequency. A spectrogram, visualised in Figure 3.1c, on the other hand, is a two-dimensional representation of the frequency content of a signal over time. The STFT is used to create it on a signal that has been windowed, with the STFT magnitude squared producing the signal's energy or power content at each frequency for each time window. With time on the x-axis, frequency on the y-axis, and color depth representing the magnitude or power of the signal at each time and frequency, the spectrogram visualises this data as a color-coded image.



(a) EEG signals with varying amplitudes over time



(b) EEG signals in the frequency domain



(c) Spectrogram representation

Figure 3.1: EEG signal representation in raw data time series, frequency domain and spectrogram representation

The raw EEG data is transformed, before being processed by the proposed deep learning framework. Firstly, the signal is pre–processed with filtered values between 2-40Hz. These frequency parameters are widely adopted when looking for EEG features. Each 5 seconds ($5 * 100Hz$) of EEG signals was sampled and converted to a spectrogram image in accordance with the time sequence. Additionally, to maximise data utilisation, each STFT window overlapped by 20% with adjacent windows. Finally, each generated spectrogram image was resized to 256 x 256 pixels in order to conform to the deep CNN's input format. Finally, the data is classified into right and left motor imagery data. Figure 3.2 shows the block diagram of the framework proposed.

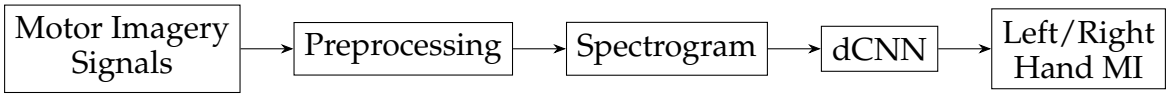


Figure 3.2: MI Classification Pipeline using STFT and deep learning framework

3.2 Data Preprocessing

The first stage of pre-processing was to reduce power line interference by applying a notch filter with a frequency of 50Hz. Second, the data was high-pass filtered with a cut-off frequency of 0.5Hz to eliminate baseline drift. The signals were filtered between 7Hz and 14Hz, after the experiment, using a 5th order zero-phase Butterworth filter.

The Nolan et al. [168] method was chosen to detect and eliminate EOG and EMG signals from the EEG data. EEG artefacts were removed from individual channels within single epochs using the channel data's mean amplitude and standard deviation. The deleted portions and epochs comprised data with a high amplitude, which was typically evaluated as per the standard. To enhance the representation of partial temporal invariance in the data and minimize overfitting and associated errors, an overlapping sliding window technique was employed.

A consensus was reached regarding the characteristics of the data windows, which should exclusively pertain to the MI and encompass the entire duration of the trial, starting from the initiation of MI until its completion. A time window of 4 seconds was used to generate alternative crops with a stride of 125 ms, creating 25 new sub-trials from each individual trial. Crops were collected for 3 seconds prior to the initiation of MI and continued until the end of the trial.

Following the deployment of the augmentation technique, a considerable performance improvement was noticed with the recommended model, as it required the CNN model to learn complex characteristics from all of the windows, resulting in enhanced classification performance. The augmentation procedure was performed exclusively on the training set. The augmented dataset also includes the original EEG signals.

3.3 Model Implementation

CNNs have been most successfully employed for image identification applications, requiring a two-dimensional input. EEG signals, by contrast, are dynamic time series obtained from several scalp channels. To address this issue, the employed EEG data is in the form of a two-dimensional matrix of size TxC, where C denotes the number of electrodes (channels) and T denotes the EEG signal time points, as represented in Figure 3.3. As illustrated in Figure 3.4, the proposed CNN model architecture is inspired by CNNs used in the ImageNet competition, such as VGGNet [169] and AlexNet [170, 126].

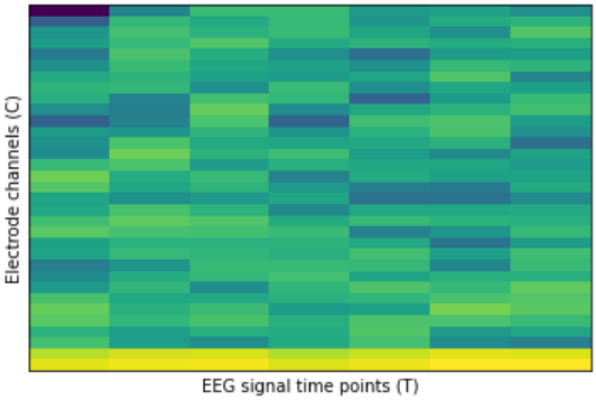


Figure 3.3: Representation of the two-dimensional matrix which represents the number of electrodes vs the time points

Determining the coarse structure of the DNN is one of the most difficult problems in DNNs. A DNN structure with a small size makes it easier to search for a coarse model than a DNN structure with a large size. It employs stacked convolutional layers with decreasing layer depth and an increasing number of filter kernels. Batch normalisation is applied after each convolutional layer to reduce covariate shift in intermediate representations and improve robustness.

A block diagram representation of the STFT model is shown in Figure 3.4. The first convolutional layer computes the spectrogram representation, which is used as the model's input. The CNN's pooling layer conducts a non-linear down-sampling operation while also contributing to the network's translation invariance. By placing a pooling layer between the convolutional layers of the CNN, the CNN's input size is gradually lowered.

Max-pooling is the most often used pooling method in ML applications. The requirement for pooling layers varies by problem, and there is no universal methodology. Pooling is a technique that is frequently employed in CNNs, but it is not required to utilise this layer after each convolutional layer. Between each subsequent convolutional layer in this investigation, a pooling layer is used. With an activation function of $f(x) = max(0, x)$, the ReLU layer zeroes out the negative values in the activation maps, performing a non-linear action in the decision-making system. Additionally, between each layer, the batch normalisation and

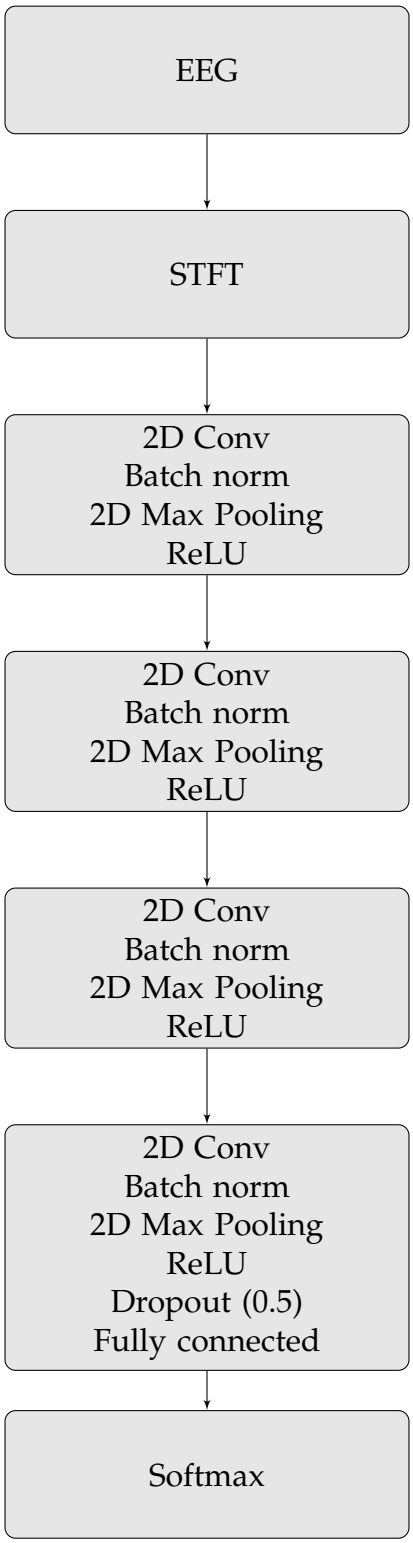


Figure 3.4: Block diagram of motor imagery classification using STFT and deep learning framework

drop-out functions are employed to reinforce the training. While a CNN represents the features in general, a fully connected NN represents the classifier in particular.

The magnitude of spectrum images was determined using STFT at a sampling rate of $250Hz$ and a 50% overlap. The spectrogram illustrates the STFT results visually. Following their acquisition, the EEG signals were initially cut into small overlapping time—windows. This new data format combines information about frequency, time, and electrode position. Due to the unique design of the Hamming window, it was chosen for this experiment as the sliding window. Following that, each crop was subjected to a FFT, with the assumption that the crops would remain stationary over a short time period. The spectrogram was generated by merging three distinct spectrograms of three electrodes from the motor cortex, C3, C4, and Cz. The spectrograms in a frequency range of 2–40Hz were used to generate the input image for the proposed CNN framework.

To avoid mains interference, a suitable notch filter was used, which varied according to the country from which the dataset was collected ($50Hz$ for the majority of European countries, $60Hz$ for the majority of others). The parameters for this experiment were $n = 1024$ FFT samples and a $s = 16$ time shift between STFT windows.

The CNN model was trained, validated and tested on each individual subject of the dataset on overall accuracy. Further to this, the model was additionally validated using a 5–fold k–means cross–validation. Different approaches to network architecture design were investigated to determine their effect on network performance. We sought to gain insight into the network by varying design parameters such as activation functions, kernel size, and filter size. Taking into account computational expenses and the time required for training the model, we opted for a simpler CNN model construction. The filters were aimed to capture different spatial pattern related to MI. The proposed dCNN model consisted of a 6-layer model. Firstly raw EEG signals were used in order to create spectrograms using STFT. The resulting images were passed into the CNN. The network has 5 convolutional layers. The filters were chosen such that the first filter is 24, and doubles for the two following $Conv2D$

layers i.e. 24, 48, 192, 384 and finally 768. A kernel size was chosen which dropped in size with each $Conv2D$ layer. The network's performance was tuned by using filters of various sizes, however, we observed the highest performance improvements with an increasing filter size, and a decreasing kernel size.

The concept of changing the filters for each convolution was chosen to emulate state-of-the-art CNN models such as the VGG or ImageNet model structure [126]. After each convolution layer a $ReLU$ activation layer was applied. The activation function had the greatest impact on the network's performance. For instance, the ReLU produced the most accurate results. Additionally, dense layers were implemented without activating the ReLU, resulting in a decreased recognition rate. This leads to the conclusion that non-linearity makes dense layers more robust at the network's end in terms of feature extraction.

To perform the classification of the data, a fully connected layer is applied further to the three convolutional layers, with a $Softmax$ activation layer. Max pooling was applied after each layer, as research indicates that this could improve the network performance. A dropout regularisation is applied after the final convolutional layer at 50%. The chosen dropout was the optimal dropout found during hyperparameter tuning to prevent overfitting. During model compilation $categorialcrossentropy$ was the chosen loss function, and we used the $Adam$ optimal optimiser.

3.4 Analysis and Results

As stated in the previous chapter, a CNN is a DNN composed of an input layer, many convolutional and pooling layer pairings, a fully connected layer, and finally an output layer. The architecture of a CNN is a top priority for DL researchers. There are various state-of-the-art CNN models for image identification, including AlexNet and ResNet. Each of these CNN models is unique in terms of layer count, kernel size, and inception blocks, which are responsible for the CNN model's classification accuracy.

Transfer learning is a ML technique that entails reusing a trained model that was created for a certain job in order to classify another task using the learned knowledge. In this section, we employed the transfer learning method to compare our CNN built on the STFT framework to other state-of-the-art models. Transfer learning of these pre-trained CNN models begins by adapting the network's structure and learnt features to the new dataset, rather than training the network from scratch using random weights. The next sections examine the network architecture of these models.

AlexNet and ResNet50 were utilised as pre-trained models used for benchmarking. Originally, these networks were created to classify photos for the ImageNet challenge, with varied degrees of classification accuracy.

Five convolutional layers and three fully linked layers comprise the AlexNet model. Each convolutional and fully connected layer output is routed through a non-linear Rectified Linear Units (ReLU). The final fully connected layer is fed into the output softmax layer, which classifies the input images using the 1000 class labels specified in the ImageNet classification challenge. The input layer receives an image with a dimension of 224x224x3. AlexNet's detailed design is described in depth in [126].

ResNet-50 is a 50-layer residual CNN that is aimed to alleviate the problem of vanishing gradients during backpropagation by introducing shortcut links between conventional CNN networks. ResNet50's input layer accepts an image with a dimension of 224x224x3. Each convolutional layer is followed by batch normalisation and downsampling using filters of size 33. To classify the input image into 1000 labels, the last component of the CNN uses an average pooling layer and a fully connected layer with softmax. ResNet50's detailed design is described in length in [129].

We chose to conduct these tests using the BCI Competition III dataset IIIa, which contains four distinct classes of motor imagery (left hand, right hand, foot and tongue). This dataset was used to evaluate both a two-class (left and right hand) and a four-class dCNN on it. The dCNN was applied to the data after each participant's spectrogram was pre-processed

using the spectrogram framework. To reduce the data's dimension and improve learning accuracy, the data was transformed as described in the pre-processing section.

In the study, a deep learning model was employed on a per-subject basis within the dataset, incorporating all EEG trials as part of the framework. A stratified five-fold cross–validation was applied for both the proposed framework and also the two state-of-the-art datasets (ResNet and AlexNet). Five folds were used for training and validation (90% training and 10% validation), and the last fold used for testing. Following the validation of 10% of the training set, the remaining data in the training set is supplemented. As a result, no augmentation of validation on test data is performed. The small initial dataset is enhanced first, and then divided into 82 percent training, 8% validation, and 10% test sets. The dCNN would have complete information about the data because the three sets of training, validation, and test would contain augmented variants of the original data. In this case, success rates of 100 percent would be achieved for all open-source datasets' training, validation, and test sets.

The inability of some individuals to participate adequately in MI-based studies is a frequent occurrence, and as a result, unsatisfactory results with specific participants are obtained. When compared to currently used methods, it is discovered that the proposed methods achieve a marginal improvement in classification performance. Each dataset was cross-validated five times. Additionally, to achieve the highest degree of accuracy, a grid search method was used to determine the optimal parameters. An early–stopping feature was used to prevent overfitting. This ensured that the model would be trained until the minimum validation loss was found and tested on the test data to measure its generalisation capabilities. Obtained results from the proposed CNN model have outperformed state–of–the–art–results on the same dataset, which were obtained using classical ML techniques.

A confusion matrix is a frequently used statistical technique for classifying situations. It is relevant to problems involving both binary and multi-class classification. The confusion matrix depicts the distribution of all expected responses in relation to their true classifications.

The following figures present the performance indicators for our dCNN on the BCI III

dataset IIIa dataset for a binary class classification model. Looking at the confusion matrix for one of the k-folds, as illustrated in Figure 3.5, the true labels are predominantly correctly identified for both the left and right classes in each of the five instances. This particular fold shows a classification accuracy of 93.12%. The classification accuracy over the cross validation folds indicates that the model is capable of learning and discriminating between the two classes on a consistent basis of left hand and right hand MI.

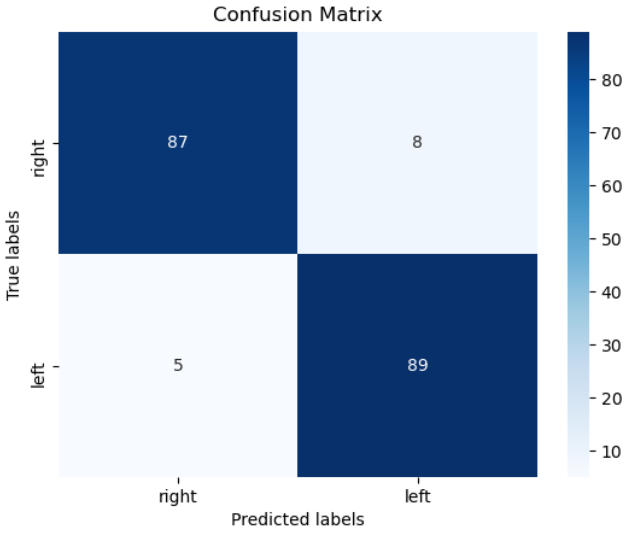


Figure 3.5: Confusion Matrix for spectrogram-dCNN framework: BCI IIIa binary classification (k3b)

Figure 3.6 illustrate the training versus validation accuracies. A significant difference in accuracy or loss between training and validation strongly suggests overfitting.

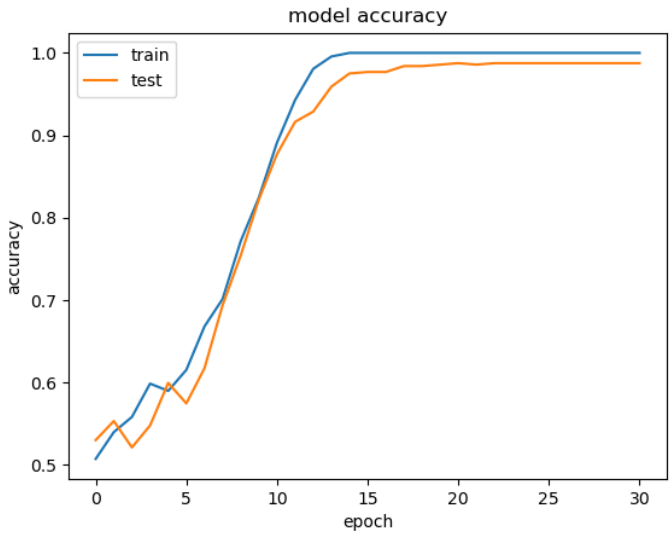


Figure 3.6: Model Accuracy for spectrogram-dCNN framework: BCI IIIa binary classification (k3b)

The larger the distance, the more overfitting occurs. As illustrated in the image, we first permitted model training to take place across 30 epochs. Both training and validation curves closely track each other for accuracy and loss, but begin to diverge at 10 to 15 epochs. This is true for all k-folds. As a result, in order to avoid overfitting, we choose 12 epochs as the optimum cut-off point for model training.

Table 3.1 summarises the classification accuracy of STFT pre-processing steps, as described previously in this chapter, followed by either the proposed model, AlexNet or ResNet50 models. As a result of these findings, it is clear that the suggested model outperformed the transfer learning approach with a pre-trained network in terms of classification accuracy.

Table 3.1: Classification results of two-class BCI competition III dataset IIIa compared to transfer learning from state-of-the-art models

	Classification accuracy %		
Subject	Proposed	AlexNet	ResNet50
k3b	94.01	90.32	89.25
k6b	91.86	87.69	86.33
I1b	89.56	86.33	87.98
Mean ± Std Dev	91.81 ± 2.23	88.11 ± 2.03	87.85 ± 1.46

It can be observed that the proposed framework has a mean classification accuracy of 91.81%, whereas AlexNet and ResNet50 both have 88.11% and 87.85% classification accuracy, respectively. This demonstrates that, despite the fact that both AlexNet and ResNet50 have a significantly higher layer count, the layer count is not positively linked to classification accuracy, and was not able to generalise over the individual participants in the dataset with a higher accuracy.

The following graphs illustrate our deep convolutional neural network's performance indicators on the BCI III dataset IIIa for a four class classification model, when applied on individual subjects within a dataset. Each of the five k-folds has its own confusion matrix. As seen in Figure 3.7, one of the k-folds, true labels are mostly accurately detected for both the left and right classes in each of the five occurrences. Classification accuracy for this specific fold is 92.06%.

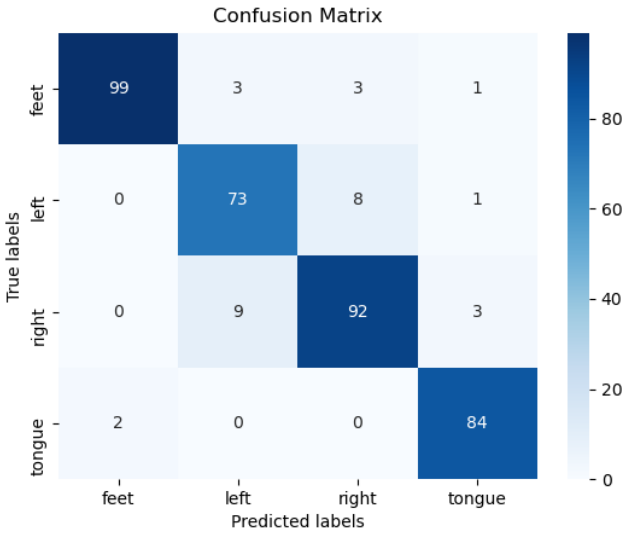


Figure 3.7: Confusion Matrix for spectrogram-dCNN framework: BCI IIIa 4-class classification (k3b)

The classification accuracy throughout cross validation folds demonstrates that the model is capable of accurately learning and differentiating between the four classes of left hand, right hand, tongue and feet MI.

The training vs validation accuracies and losses are depicted in 3.8 and 3.9 for the four class classification of this dataset. Between training and validation, a considerable difference in accuracy or loss clearly signals overfitting. Initially, the model was run for 30 epochs. From the model accuracy and loss curves we are able to visualise that the point where the model appears to be overfitting is approximately at 15 epochs. As a result, we chose 15 epochs as the optimal cut-off point for model training to avoid overfitting.

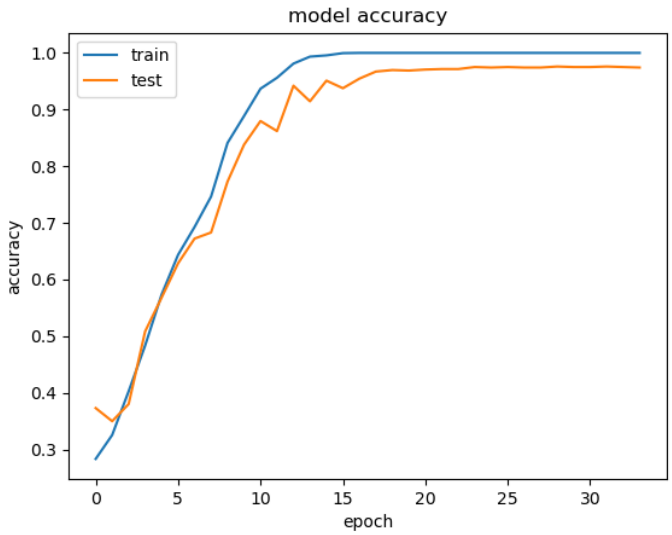


Figure 3.8: Model Accuracy for spectrogram-dCNN framework: BCI IIIa 4-class classification (k3b)

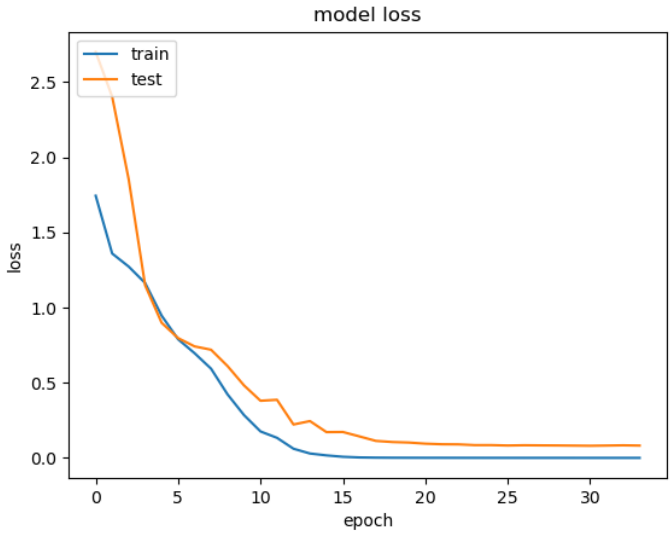


Figure 3.9: Model Loss for spectrogram-dCNN: BCI IIIa 4-class classification (k3b)

Following the successful application of the STFT-dCNN framework on individual subjects, it was deemed appropriate to proceed with the analysis of the four-class dataset across individual subjects. The Table 3.2 summarises the classification accuracy of STFT pre-processing

steps followed by a dCNN. The dataset was additionally compared with the AlexNet, and ResNet50 networks, similiar to the binary results. As can be shown, the suggested framework has a mean classification accuracy of 90.81% , while AlexNet and ResNet50 both have classification accuracy of 86.28% and 85.11%, respectively.

Table 3.2: Classification results of four-class BCI competition III dataset IIIa compared to transfer learning from state-of-the-art models for spectrogram-dCNN framework

	Classification accuracy %		
Subject	Proposed	AlexNet	ResNet50
k3b	92.98	87.22	85.34
k6b	89.22	85.39	83.45
I1b	90.23	86.24	86.55
Mean ± Std Dev	90.81 ± 1.94	86.28 ± 0.92	85.11 ± 1.56

Following the successful application of the STFT-dCNN framework on the individual subject 4-class classification in the dataset, it was decided to evaluate this framework across the entire dataset.

It was observed that the proposed framework has a classification accuracy of 74.97%, as detailed in Table 3.3. This accuracy is well below what would be considered an acceptable level of classification accuracies for BCI MI. However, this level of accuracy is also exhibited when utilising transfer learning and the spectrogram framework with AlexNet and ResNet50, which achieve classification accuracy of 72.02% and 69.68%, respectively.

Table 3.3: Classification results of two-class BCI competition III dataset IIIa compared to transfer learning from state-of-the-art models for the overall dataset

	Classification accuracy %		
k-fold	Proposed	AlexNet	ResNet50
1	74.47	73.62	70.22
2	77.42	74.55	69.58
3	71.99	70.86	72.33
4	72.14	71.83	67.28
5	73.82	69.22	68.98
Mean ± Std Dev	74.97 ± 2.20	72.02 ± 2.13	69.68 ± 1.84

3.5 Conclusions

In this research a DNN based model on pre-processed spectrogram EEG data was proposed. In order to classify the MI signals without the use of identifying specific features the dCNN was used. The concept of feature vectors in the time-frequency domain was proposed and refined as an effective and practical framework for removing noise, reducing dimension, and extracting signal features from EEG data.

The following factors may contribute to the proposed framework's superior performance: To begin, the spectrogram structure emphasises the characteristics of each class. Then, using the suggested DL system, the spatial and temporal properties of the MI signal may be efficiently learned. In comparison to existing frameworks, the experimental results reveal that the proposed scheme is extremely effective at enhancing discriminant analysis results and also has a strong classification effect on MI signals with short time series.

We proposed a unified, end-to-end classification system that integrates two stages: spectrograms and dCNNs. The spectrogram framework is effective in highlighting characteristics and assisting our proposed NN in achieving greater accuracy in MI task categorisation. The

disadvantage of this approach is that the model is applied to each individual participant rather than to the entire dataset. We discovered that the total dataset performed poorly, and the dCNN with the spectrogram architecture was unable to generalise across it.

In comparison to the well-established ML approaches highlighted in the literature (Chapter 2), this framework appears to be superior at discriminating MI task-related EEG signals. It outperformed other classification algorithms by achieving the greatest classification accuracy of 91.81% for binary classification and 90.81% for four-class classification. It outperforms a number of other notable combinations of feature extractors and classifiers. Thus, combining STFT modifications for EEG signal reconstruction with dCNN models proves to be a robust approach when applied on individual subjects.

4

Deep learning using topographical EEG images for classification of user intention

This chapter examines the combination of topographical maps with a dCNN as an alternative candidate for EEG MI classification. Topographical maps provide a spatial representation of a window of information, effectively making them images that can be used in conjunction with convolutional networks. The following chapter presents our proposed approach as an alternative framework to address the cross-subject generalisation problem observed in the previous chapter.

4.1 Introduction

Constructing a model that is robust to signal translation and deformation in space, frequency, and time due to inter-subject and intra-subject variances, as well as signal acquisition techniques, is a fundamental problem in correctly recognising mental processes from observed brain activity [171]. Many of the discrepancies are due to minor individual changes

in cortical mapping and/or functioning, which results in observable disparities in spatial, spectral, and temporal patterns. Furthermore, lack of proper fitting of the cap on heads of various sizes and shapes, EEG caps, which are used to position the electrodes on top of designated cortical regions, might be another source of spatial differences in observed responses.

We propose an improved DL based framework for learning representations from EEG data, that appears to be more robust to inter-subject and intra-subject variations, as well as measurement related noise. The method proposed in this thesis is substantially different from prior deep neural network based attempts to learn high—level representations from EEG. Instead of representing low–level EEG features as a vector, we transform the data into a multi-dimensional tensor which preserves the data's structure throughout the learning process. In other words, unlike current EEG analysis approaches, which neglect such spatial information, we produce a sequence of topology-preserving multi-spectral images.

Following the extraction of these topologies, the EEG data is trained on DNN architectures influenced by state–of–the-art video classification techniques in order to develop robust representations from a succession of images, or frames [172]. CNNs are used to extract spatial and spectral invariant representations from each frame of data [173].

In literature, researchers generally train EEG MI models on a subject-by-subject basis i.e. the model would be applied on the individual's trials rather than applied between participants. Applying subject-by-subject training restricts the use of your data. DNNs are capable of generalisation and classification based on the amount of properly labelled data at their disposal. As stated previously in this thesis, this project's scope is extremely constrained by lack of available open-source datasets.

In contrast to previous research that has focused on subject-by-subject comparisons, the work in this chapter will examine the possibility of developing a general classification model that is capable of generalising across all subjects. As detailed in the literature review chapter, the framework made use of a collection of publicly available datasets. Experiments with

two, three, and four classes of MI were conducted. Section 2.11 provided an in-depth description of each dataset.

4.2 Data Preprocessing

EEG data consists of numerous time series corresponding to measurements taken at various spatial locations throughout the cerebral cortex. Like voice signals, the most salient aspects of a signal are found in the frequency domain, which may be analysed by looking at the signal's spectrogram. Nevertheless, as previously mentioned in the literature review, the EEG signal possesses a spatial component. A FFT is performed for each trial to evaluate the power spectrum of the signal. Cortical oscillations which are associated with frequency bands for memory processes are theta (4-8Hz), alpha (8-13Hz), and beta (13-30Hz). Calculated and used as a distinct measurement for each electrode, the sum of squared absolute values within each of the three frequency bands was computed and used as a separate measurement.

The standard approach to EEG analysis is to aggregate spectral measurements from the electrodes to form a feature vector. This approach, however, ignores the inherent structure of EEG data in terms of space, time and frequency. The proposed framework preserves spatial structure by transforming the measurements into a 2D image, using multiple colours to represent the spectral dimension. Temporal evolutions in brain activity are represented from a sequence of images from consecutive time windows.

Each of the EEG trials can be expressed mathematically as a matrix, $T \times i \times j$ of $M \times N$, where i represents the index of the MI tasks, and j represents the index of trials. M expresses the number of channels in the EEG, and N represents the time–domain visual matrix. In order to preserve the relative distance between the electrodes which are normally on the cap, Polar Projection is applied to 3D coordinates which projects this onto a 2D surface. This procedure generates a 32×32 image. The process is then repeated for each of the frequency

bands discussed, and merged, giving an image which is the collective representation of the three frequency bands.

This three-channel image gives the input to the deep CNN. The data pre-processing can be segregated into four steps. Firstly, The data is separated into overlapping one–second windows, and the Hanning window is applied. Second, an FFT is applied to the data to convert the data from the time domain to the frequency domain. Thirdly, the FFT amplitudes are grouped into alpha(8–12Hz), beta(12-40Hz), theta (4–8Hz) frequency band. This would then give three quantifiable values for each of the electrodes. Finally a mathematical projection tool, the 2D azimuthal projection is applied to the values produced by the frequency binning are then interpreted as RGB colours and projected onto a 2–D map of the head. We go into detail of these steps over the next few paragraphs.

The Hanning window was applied on the raw signal as the first part of the data pre–processing. The main lobe width of the Hanning window is $\frac{8\pi}{T}$, and roll–off rate at $-18dB/oct$, and side load level at $-31.5dB/oct$. The equation can be seen in 4.1, where n is the sample index, ranging from 0 to $N-1$, and N is the length of the window function. Generally, the Hanning function is the chosen windowing function due to its good frequency resolution and reduced spectral leakage.

Hanning function:

$$w(n) = 0.5\left[1 - \cos\left(\frac{2\pi n}{N-1}\right)\right] \tag{4.1}$$

Hanning windows are frequently used with random data because their effect on the frequency resolution and amplitude accuracy of the resulting frequency spectrum is relatively small in comparison to the effects of other windows. A Hanning window's maximum amplitude error is 15%, while the frequency leakage is typically limited to 1.5 spectral lines on either side of the original sine wave signal.

The Hanning window begins at zero and terminates at zero. It has a value of one in the window's centre. This gradual transition between 0 and 1 ensures a smooth change in am-

plitudes when the measured signal is multiplied by the window, thereby reducing spectral leakage.

DFT is a mathematical technique used to transform a discrete signal from its time domain representation to its frequency domain representation. The DFT converts a discrete signal into a sum of complex exponentials, each representing a particular frequency component in the signal. The magnitude and phase of each frequency component can be found from the DFT coefficients, which are complex numbers.

FFT is an efficient algorithm for computing the DFT of a signal. It reduces the computational complexity of the DFT from $\theta(n^2)$ to $\theta(n \log n)$, where N is the number of samples in the signal. The FFT algorithm exploits the symmetry properties of the DFT to reduce the number of computations required, making it much faster than the standard DFT. The FFT is widely used in many applications, including signal processing, image processing, and communication systems.

The EEG data is recorded from the scalp surface. This data contains two types of information: voltages and electrode positions. There are three commonly used methods for displaying data. The most common method is to use a set of lines known as isolines, each of which represents the same voltage values on the head surface. However, it cannot show us the true positions of the maximum and minimum because the interpolation method used may give us an incorrect impression of brain activity.

The second method, using potential distribution on a projective plan, can provide us with the voltages and 2-D positions, but there is still the issue that when a hemisphere is projected to a plane, there is some serious distortion in the projection of the sphere's lateral surface. The final step is to place the data on the realistic head surface. However, in order to be analysed, this model must be rotated. The azimuth projection has long been used to represent EEG data on maps. To convert the data from its raw form, three steps were taken. First, we convert the electrode coordinates to 'geographical' coordinates. Second, we use the azimuth conformal projection to determine the new electrode position. Finally, the

interpolation method is used to obtain a satisfied figure. Figure 4.1 provides a sample of four images post pre–processing using the framework above. It is observed that the spatial structure is preserved. In Figure 4.2 the complete data pipeline is represented. The raw EEG data is fed into the data pipeline using the pre–processing framework as above. The generated EEG temporal representations are then fed into a deep CNN model architecture. The outcome of the model is the directional prediction, depending on the dataset this would be either two, three or four classes.

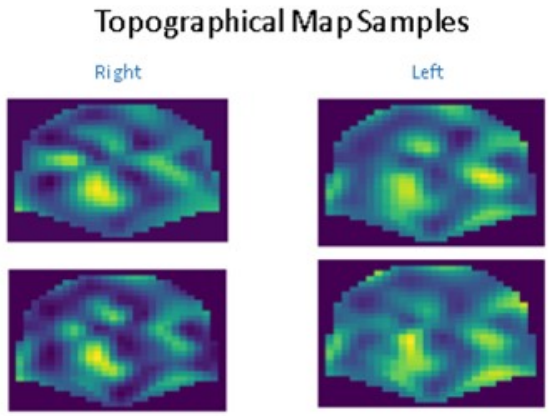


Figure 4.1: A sample of the produced 2–D azimuthal projections

4.3 dCNN Model Implementation

Deep CNNs lie at the core of state–of–the–art architectures when working with image and video data. EEG data when presented is in the form of X number of channels of data, against the y data samples taken, as a 2D–Matrix of data. The process of this experiment is based off the approach taken by Bashivan et al, [174]. The data which the model is based on was collected with the intent of classifying the mental exertion required to count different number of letters on a screen.

EEG signals are time–based, they require to be transformed appropriately to be compatible with CNNs. Each of the EEG trials can be expressed mathematically as a matrix, T_j^i of di-

mensions $M \times N$, where i represents the index of the tasks, and j represents the index of trials. M expresses the number of channels in the EEG, and N represents the time–domain data points.

Bashivan et al. [174] uses general the VGG network framework [169], popularly known for its object detection and segmentation within images. All the convolutional layers are 3x3, and stride length of 1 pixel, max pooling at 2×2, and stride of 2 pixels, using a ReLU activation function, with fully connected layers at the end, also using a ReLU activation function. Image margins are padded in order to preserve the whole image between convolutions. Multiple convolutional layers are stacked before being followed by a max pooling layer. As we go deeper into the network, after each of the max-pooling layers, the kernels are increased by a multiple of 2.

Initial evaluations were conducted on a 5 layer CNN model without any preprocessing of the EEG data. Raw EEG signals were used from all 64 channels provided in the database [160]. Spectrograms were generated from the EEG signals and the resulting images were fed to the NN as inputs of the CNN, and was then passed through two densely connected layers for classification using a sigmoid activation layer. The model was consistently underperforming with results at around 40% testing accuracy. While hyperparameter tuning was able to slightly increase the accuracy of the model, it had no significant impact. Thus, this avenue was abandoned from further investigation, as the model was unsuccessful in learning the features from raw data.

The successful implementation proposed in Figure 4.2 consists of a 9 layer neural network and a four stage preprocessing step of the EEG data. The preprocessing steps described in Section 4.2 (Hanning Window FFT, Frequency Binning) were applied to the raw data. The final stage of the preprocessing pipeline consists of generating the topographical maps by 2D Azimuthal Projection. These images are then being fed to the neural network through two convolutional layers and alternately max-pooled. For the intermediary transitions between convolutional layers, ReLU was applied as the activation function. The resulting matrix is

then flattened and a dropout is applied before passing through a dense layer to calculate the probabilities of it belonging to a particular class. For the optimisation, the Adam was used. The large amount of parameters in EEG datasets makes them susceptible to overfitting. In order to mediate this risk, a dropout with probability rate of 0.1 was applied to all fully connected Layers. Learning rate was set to 10^{-3}, with a batch size of 32. Training was carried out using the cross-entropy loss function. A similar VGG structure was used, with modifications made to the complexity of the original structure. Optimisation search tools were implemented, in particular using Grid–search based method using the Talos library [175]. To reduce training time for models, early stopping was also applied.

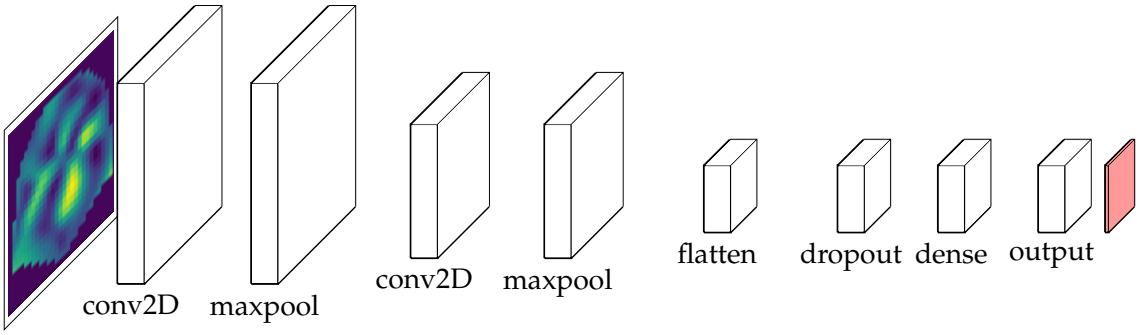


Figure 4.2: Proposed dCNN Model

The proposed CNN model can be seen in Figure 4.2. Its structure is inspired by the VGG model architecture proposed in [170]. Further to testing, the optimal found hyperparameters were as follows with $momentum = 0.9$ and $learningrate = 0.0010$. The dropout layer in the model is set to 0.1.

4.4 Performance Metrics

It is crucial to use a suitable metric when analysing ML models. For the experiments conducted in this work, performance was evaluated using a number of classification metrics. Numerous metrics have been proposed for evaluating ML models in a variety of applications. In certain circumstances, focusing exclusively on a single metric may not provide a

complete picture of the problem. In order to have a thorough understanding of a model's metrics a combination of performance metrics should be considered.

There are several methods to evaluate a classification model. The most popular metrics used for model evaluation include accuracy, precision, recall, F1-score, Receiver Operating Characteristic (ROC) and the Area Under the Curve (AUC). Classification accuracy is one of the basic metrics known; it is defined as the number of correct predictions divided by the total number of predictions, multiplied by 100.

The F1–score was the chosen measure of performance is it takes into account both recall and precision. Recall can be described as the true positive rate of the model and precision as the positive prediction value of the model. In ML, the true positive rate is either referred to as sensitivity or recall. Recall and precision can be described as equations as shown in Equation 4.2 and Equation 4.3. The resulting F1 score can be calculated using Equation 4.4. TP refers to the true positive, FN to false negative and FP for false positive. There are numerous instances where classification accuracy does not accurately reflect a model's performance. One of these circumstances is when the distribution of your classes is skewed. Confusion matrices were also generated in addition to these. They provide an instant visualisation of the performance.

$$Recall = \frac{TP}{TP + FN} \tag{4.2}$$

$$Precision = \frac{TP}{TP + FP} \tag{4.3}$$

$$F1 = 2 \times \frac{Recall \times Precision}{Recall + Precision} \tag{4.4}$$

The ROC is a graph that depicts a binary classifier's performance as a function of its cut-off threshold. It provides a visual representation of the true positive rate (TPR) against the false

positive rate (FPR) for a range of threshold values. The ROC curve provides a visual representation of the trade-off between TPR and FPR as the decision threshold is varied. AUC is a scalar value that summarises the performance of a binary classifier by measuring the area under the ROC curve. AUC provides a single, scalar metric for evaluating classifier performance that takes into account both TPR and FPR. A classifier with a high AUC is generally considered to have good performance, as it implies that it can accurately distinguish positive and negative instances over a wide range of decision thresholds.

4.5 Analysis and Results

All data analysis was conducted using a EVGA GeForce GTX 1080 Ti GPU. The Tensorflow ML framework was used alongside the Keras library as detailed in Appendix B.

A nine-layer dCNN was applied in this study, which included an input layer, eight hidden layers, and an output layer. The ReLU layer trains on the actual target class, making it a supervised method. All code was written in Python and used Tensorflow as a backend.

Numerous experiments have been conducted to develop the topographical mapping-DNN framework that has shown promise. Table 4.1 shows the results of random-search based hyperparameter selection search using Talos with using the proposed framework. For each of the frameworks described, a two-layer DNN up to a nine-layer DNN was evaluated. In this work, the aim was to keep the complexity of the system to a minimum in the hopes this work would be useful for real-time assistive device applications where the need to be less computationally expensive is paramount. This was to determine whether there was a trend in the classification accuracy of the model as the number of layers increased. If there was no positive trend in accuracy the framework was abandoned.

The first framework used band-pass filtering to transform the raw EEG data as little as possible, but this resulted in poor CNN classification accuracy results. In the second framework, we limited the data to the three known electrodes (C3, C4, CZ) that have been shown to

Table 4.1: Talos Hyperparameter search

epochs	val loss	val acc	loss	acc	momentum	lr	batch	dropout	features
80	0.12	0.97	0.06	0.98	0.9	0.0010	32	0.10	32
103	0.09	0.97	0.10	0.96	0.9	0.0010	64	0.05	32
96	0.09	0.97	0.10	0.96	0.6	0.0010	64	0.05	32
135	0.09	0.97	0.10	0.96	0.9	0.0010	64	0.05	32
150	0.09	0.97	0.10	0.96	0.9	0.0005	64	0.05	32
116	0.09	0.97	0.10	0.96	0.6	0.0010	64	0.05	32
113	0.09	0.97	0.10	0.96	0.9	0.0010	64	0.05	32
124	0.09	0.97	0.10	0.96	0.9	0.0005	64	0.05	32
134	0.10	0.97	0.14	0.94	0.9	0.0004	64	0.15	32
150	0.09	0.97	0.12	0.95	0.6	0.0004	64	0.10	32

be active during MI, while still band-pass filtering it. The third framework took a different approach, employing spectrograms and converting the data to the frequency domain, as discussed in the previous chapter, and yielded some promising results. However, the framework did not perform consistently across datasets.

A five–fold cross-validation approach was employed on the data. Four folds were used for training and validation (90% training and 10% validation), and the last fold used for testing. The ResNet model is currently considered one of the top-performing image-classification CNNs; it is short for Residual Network. To ensure consistency between comparisons, the ResNet model was also trained using the topographical maps as input. In the below sections the proposed framework classification results will be compared to the state-of-the-art ResNet model.

It was been found by [176] that when the deeper network begins to converge, a degradation problem becomes apparent: as network depth increases, accuracy becomes saturated and then rapidly degrades. This degradation is not due to overfitting, nor is it due to additional layers to a deep network that results in a higher training error. The deterioration of training accuracy indicates that not all systems are straightforward to optimise. This is aligned with experiments such as those that have been run by Narejo et al [177] and Lin et al [178].

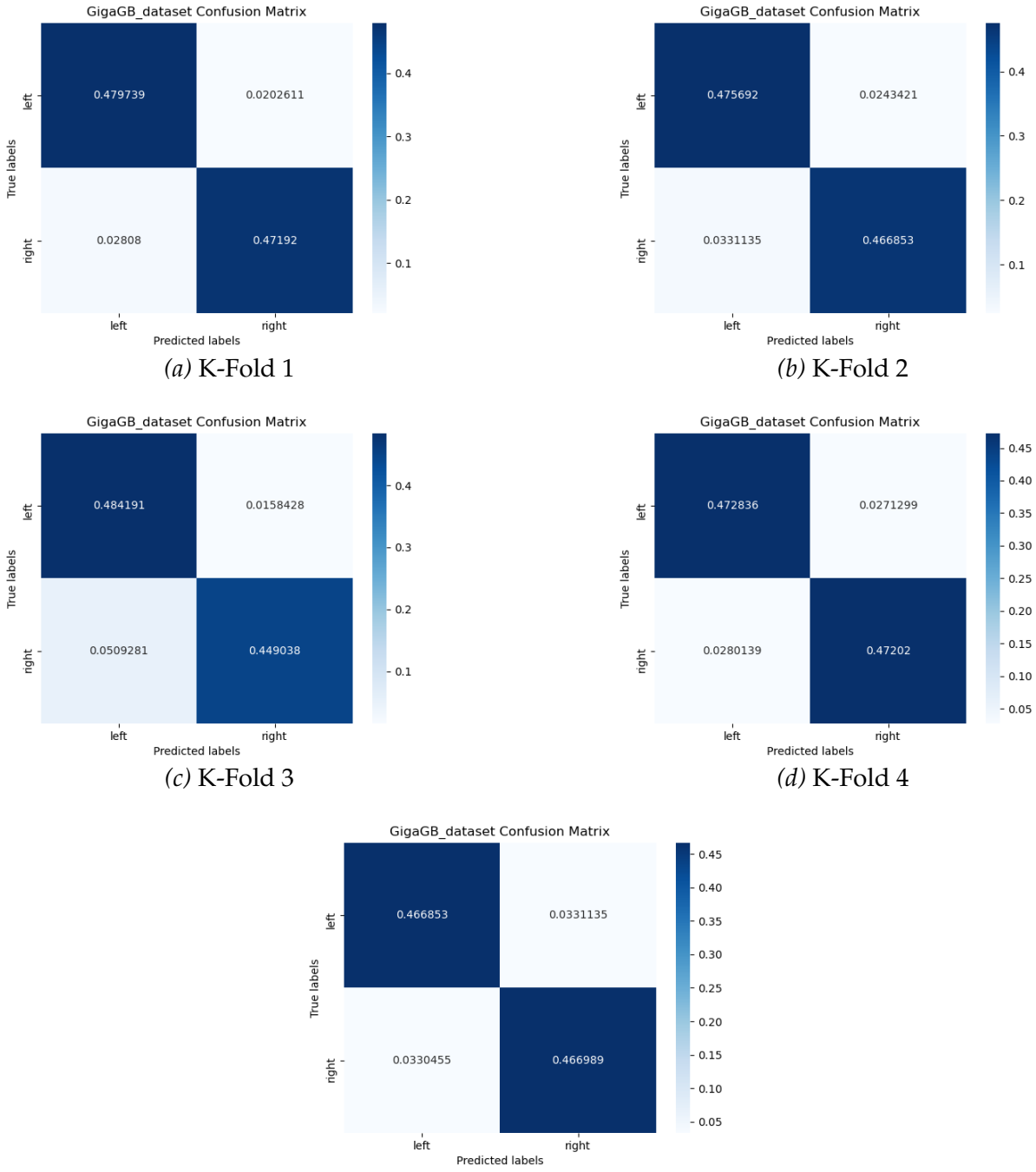
The following sections present the results from applying the proposed dCNN and topographical map framework to the datasets presented in section 2.11.

4.5.1 GigaDB dataset

The GigaDB dataset [159] has recordings of 52 subjects, as explained in section 2.11.8 of the literature review chapter.

A confusion matrix is a widely used statistic for classification problems. It is applicable to both binary and multi-class classification issues. The confusion matrix plots the distribution of all predicted responses against their true classes. Each diagonal element represents a successfully classified result. The misclassified outcomes are depicted on the confusion matrix's off diagonals. As a result, the optimal classifier will have a confusion matrix that contains only diagonal members and all other elements are set to zero. Diagonal components represent the number of points for which the predicted label is equal to the true label, and off-diagonal elements represent points for which the classifier incorrectly categorised them.

The performance metrics for our dCNN over the GigaDB dataset are illustrated in the following figures. Figure 4.3 presents the confusion matrices for each of the 5 k-folds. As can be observed from Figure 4.3, the true labels are predominantly correctly identified for both the left and right classes in each of the 5 instances, indicating that the model is able to consistently learn and distinguish the features between the 2 classes.



(a) K-Fold 1

(b) K-Fold 2

(c) K-Fold 3

(d) K-Fold 4

(e) K-Fold 5

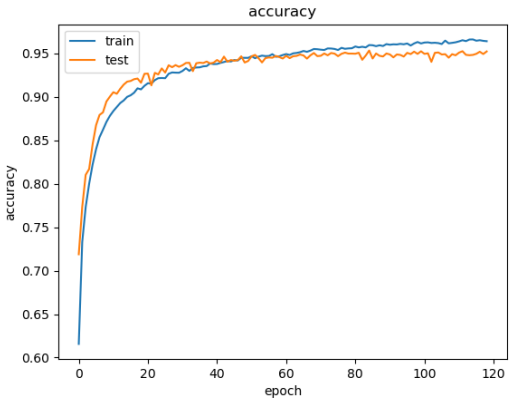
Figure 4.3: Confusion Matrix for proposed Model over GigaDB Dataset

We have chosen to train the model over 70 epochs. This has been done to mitigate the problem of overfitting. Following, we present a justification for our epoch selection in training

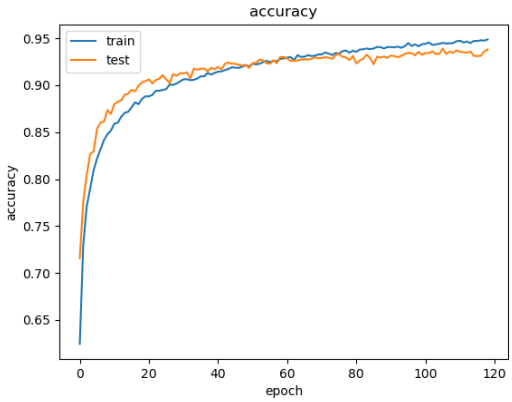
our model.

Overfitting is a term that refers to a model that has learned too much about the training dataset, including the statistical noise or random fluctuations contained within the training dataset. To fit a more flexible model, a greater number of parameters must be estimated. These more complex models may result in a phenomenon known as overfitting the data, which essentially means that they closely monitor the errors, or noise. The issue with overfitting is that as a model becomes more specialised to training data, its ability to generalise to new data decreases, resulting in an increase in generalisation error. The model's performance on the validation dataset can be used to quantify this increase in generalisation error.

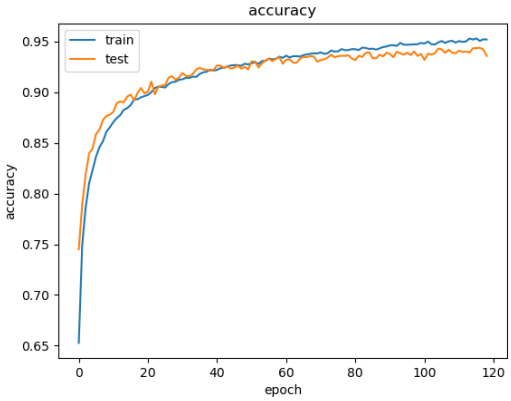
Figures 4.4 and 4.5 present the training versus validation accuracies and the training versus validation losses, respectively, for each of the 5 k-folds. A large difference in accuracy or loss between training and validation is a strong evidence of overfitting. The greater the gap, the greater the degree of overfitting. As can be observed from the figures, we initially allowed the model training to occur over 120 epochs. For both accuracy and loss, both training and validation curves follow eachother closely, but start presenting some divergence beyond 60 to 90 epochs. This effect is more subtle in the accuracy graphs but is clearly observable in the losses from Figure 4.5. At this junction, the testing metrics continue to improve but not so for the validation ones, indicating that the model becomes too specialised to generalise. This behaviour is observed for all k-folds. As such, to prevent overfitting, we decide on 70 epochs as an appropriate cut-off point to end the model training.



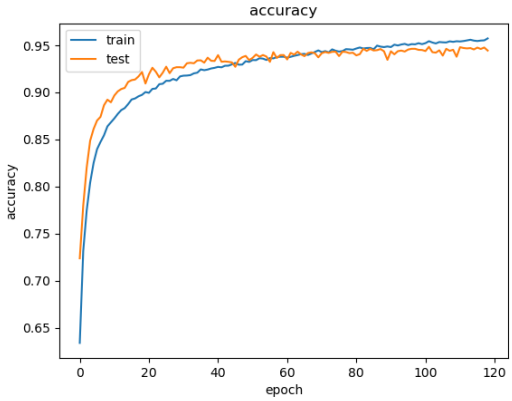
(a) K-Fold 1



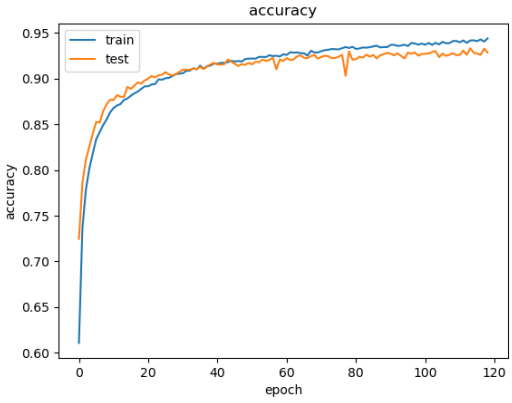
(b) K-Fold 2



(c) K-Fold 3

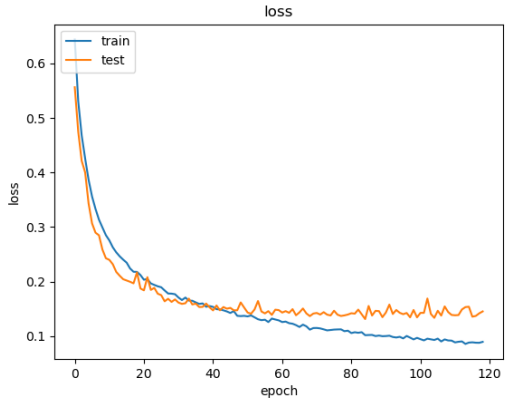


(d) K-Fold 4

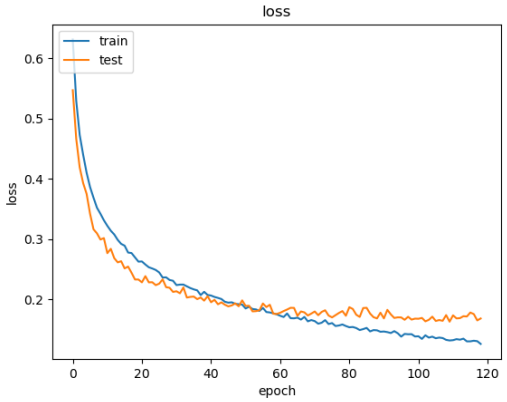


(e) K-Fold 5

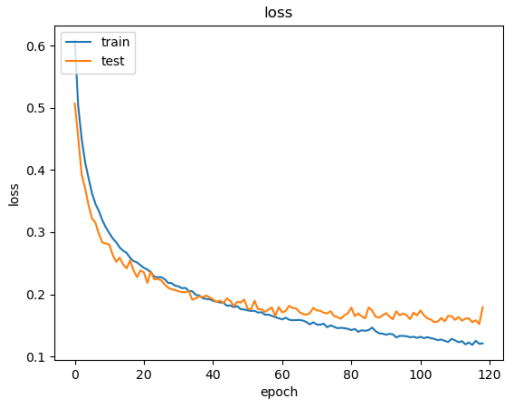
Figure 4.4: Accuracy Curves for proposed Model over GigaDB Dataset



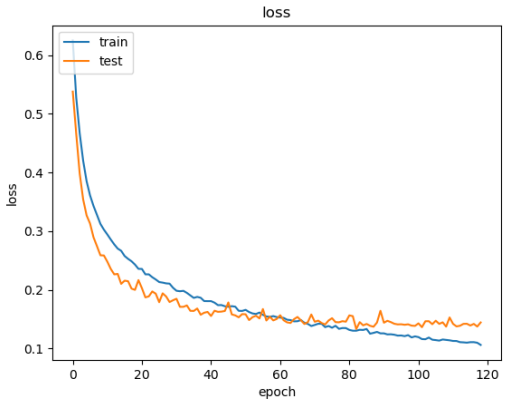
(a) K-Fold 1



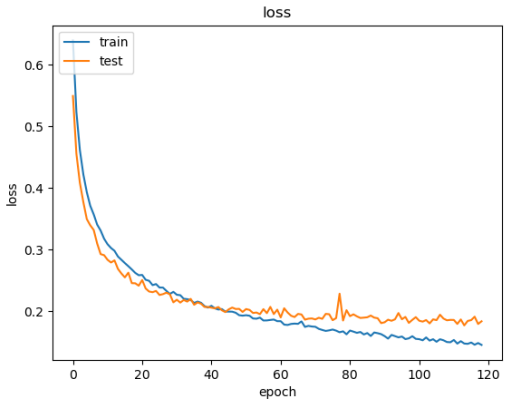
(b) K-Fold 2



(c) K-Fold 3



(d) K-Fold 4



(e) K-Fold 5

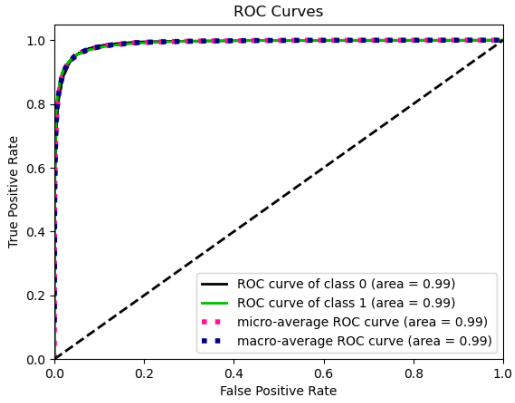
Figure 4.5: Loss Curves for proposed Model over GigaDB Dataset

The ROC curve is a binary classification issue evaluation metric. It is a probability curve that displays the TPR against the FPR at various threshold levels, thereby separating 'sig-

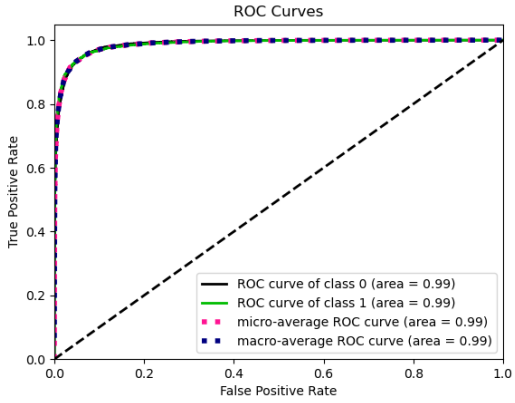
nal' from 'noise'. The ROC curve illustrates the trade-off between sensitivity (or TPR) and specificity (1 – FPR). Classifiers that produce curves closer to the top-left corner perform better. Sensitivity and specificity are two components that indicate the intrinsic validity of a result when compared to a gold standard.

As a starting point, a random classifier is supposed to produce diagonal points (FPR = TPR). The closer the curve approaches the ROC space to a 45-degree diagonal, the less accurate the model. The ROC curve is independent of the class distribution. This makes it an excellent tool for evaluating classifiers that forecast infrequent events such as diseases or natural disasters. By binarising the result, the ROC curve is tailored for multi-label classification. A separate ROC curve is produced for each class label.

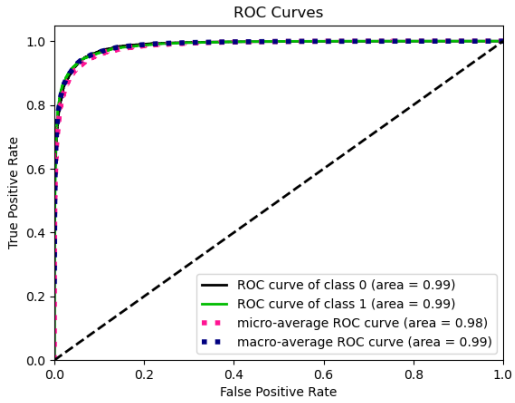
The area under the ROC curve (AUC) is a widely used indicator for summarising the information contained in the curve. AUC is a measure of performance that is aggregated across all possible categorisation levels. AUC can be interpreted in several ways. One method is to think of it as the likelihood that the model will rank a random positive example higher than a random negative example. AUC quantifies the total two-dimensional region beneath the curve in two dimensions. The AUC score indicates how well a metric can differentiate between classes. AUC is a scale-invariant and classification-threshold-invariant measure of performance across all potential classification thresholds. Figure 4.6 illustrates the ROC curves of our model over the GigaDB dataset. The model presents with AUCs of 0.99 for all five k-folds, indicating it can easily differentiate between left and right handed signals.



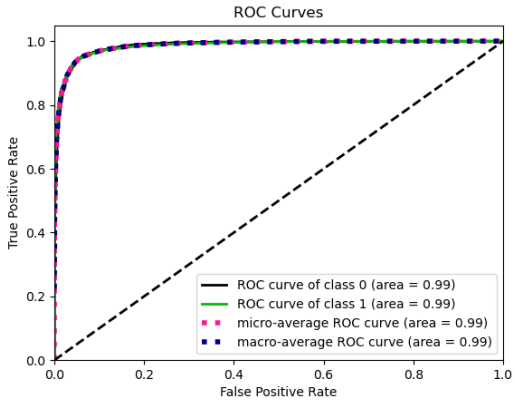
(a) K-Fold 1



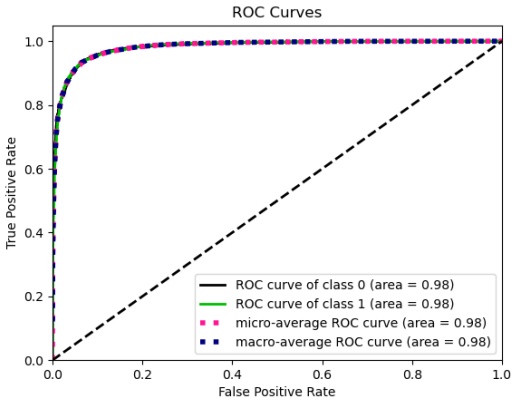
(b) K-Fold 2



(c) K-Fold 3



(d) K-Fold 4



(e) K-Fold 5

Figure 4.6: ROC Curves for proposed Model over GigaDB Dataset

Precision assesses the proportion of positive class predictions that are genuinely positive class predictions. Recall quantifies the amount of positive class predictions made from the

dataset's positive examples. We can see there is a high precision and recall within the data. There was an even split with the left and right data points which were shown to the model for training. During each of the five folds of training the AUC can be seen as 0.98 and above. The AUC is at an average of 0.99 over all the folds which indicates that the model learned to distinguish between the classes. This shows that the model is robust in generalising for the GigaDB dataset.

When target classes are not balanced, utilising the above metrics as a criterion of performance may not be sufficient. As a result, it is recommended that additional measures like as Precision, Recall, and F1 Score be considered. The findings for these metrics are given in Table 4.2. We can see by Table 4.2 that all 5 k-folds which were evaluated performed with an average testing accuracy of 95.17%.

Table 4.2: K-Fold summary for dataset GigaDB

k-fold	loss	accuracy	f1	precision	recall
1	0.15	95.17	95.12	95.12	95.12
2	0.15	94.25	94.20	94.20	94.20
3	0.18	93.32	93.25	93.26	93.25
4	0.16	94.49	94.46	94.46	94.46
5	0.18	93.38	93.37	93.35	93.36
mean	0.16	94.12	94.08	94.08	94.08
std	0.01	0.70	0.70	0.70	0.70

To assess the effectiveness of the proposed deep convolutional neural network model without requiring manual feature extraction, we chose to compare it to the ResNet50 deep network, a very popular, state-of-the-art machine learning model. The comparative experimental results obtained from running the GigaDB database through the ResNet50 network can be observer in Table 4.3.

The results obtained by our network are comparable to those obtained with ResNet50, with

Table 4.3: Resnet50 K-Fold summary for dataset GigaDB

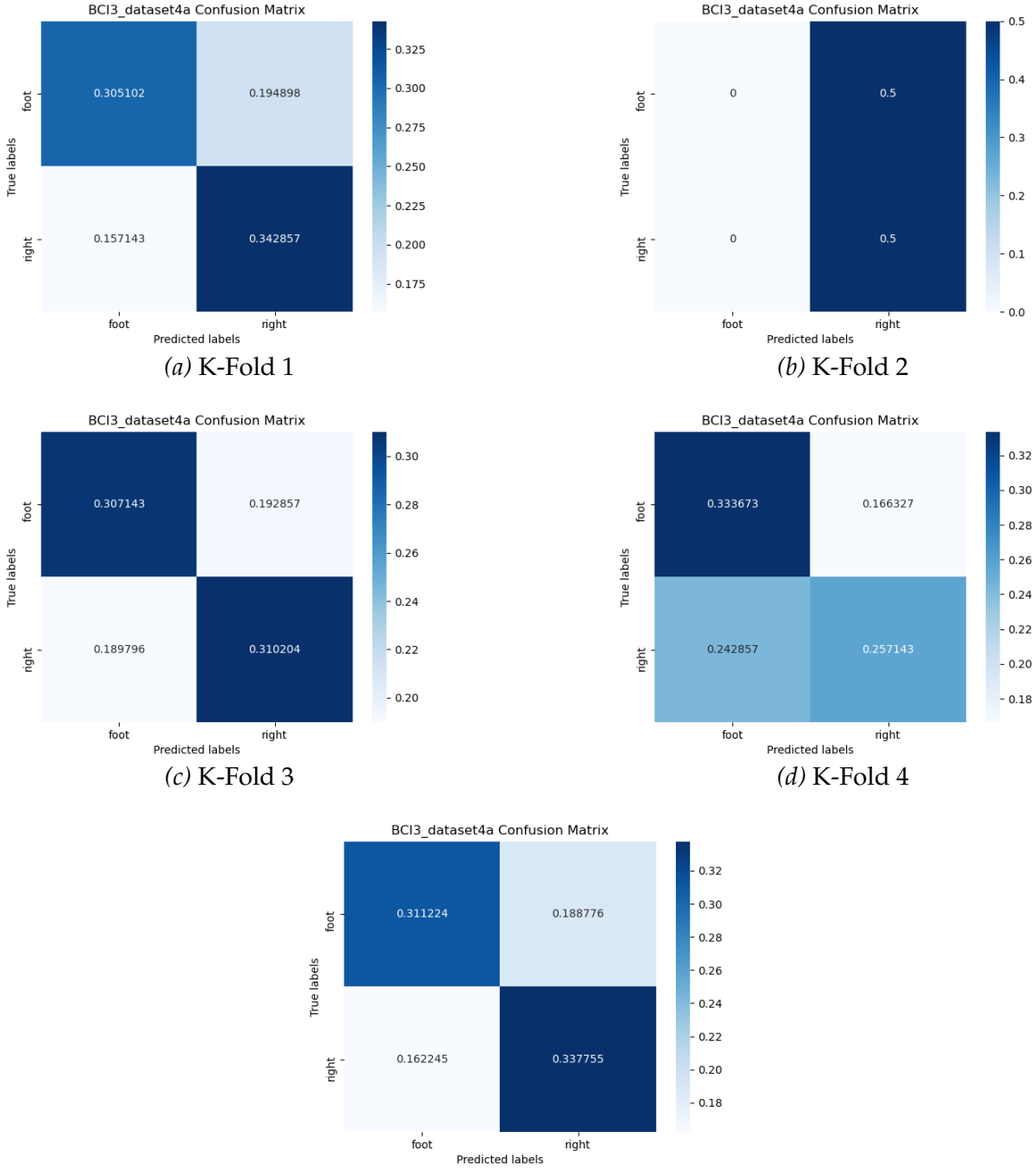
k-fold	loss	accuracy	f1	precision	recall
1	0.22	95.21	95.21	95.21	95.21
2	0.17	95.82	95.78	95.78	95.78
3	0.24	95.34	95.29	95.29	95.29
4	0.20	95.23	95.21	95.21	95.21
5	0.21	94.21	94.20	94.20	94.20
mean	0.21	95.16	95.14	95.14	95.14
std	0.02	0.52	0.51	0.51	0.51

our network slightly underperforming against ResNet50. Our model performed with an accuracy between 93.32% and 95.17% with a mean accuracy of 94.12% with a standard deviation of 0.7. The performance of ResNet50 was slightly better, with an accuracy between 94.21% and 95.82% with a mean accuracy of 95.16% and slightly lower standard deviation of 0.52, outperforming our network by 1.04% on average.

4.5.2 BCI III dataset IVA

Dataset BCI III IVa presented a real challenge for both our network as well as for ResNet50. The dataset comprises of two classes, right hand and left foot. Neither model could be successfully trained and acceptably generalise over the data. As can be observed from the confusion matrix in Figure 4.7, the model had a very difficult time distinguishing between right hand and left foot data.

While there is no clear justification or evidence in the literature regarding this behaviour, the author conjectures that a possible reason for this behaviour is the subtlety of EEG signals generated by foot MI. The motor cortex for hands has a relatively large surface area when compared to the one dedicated to foot movement. The location of the mapping for hand movement is also tangential to the cranium, which allows for EEG signals to be picked up over a wider range of the scalp. The cortical region for feet, on the other hand is positioned within the inter-hemispheric fissure separating the two hemispheres, and is perpendicular to the cranium. This would imply a smaller surface area on the scalp for the signal collection, as well as a weaker signal, and thus less information. This, however, is a theory and would require further investigation to determine the source of this issue.



(a) K-Fold 1

(b) K-Fold 2

(c) K-Fold 3

(d) K-Fold 4

(e) K-Fold 5

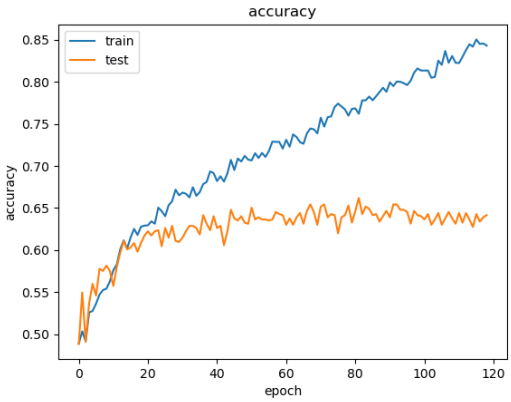
Figure 4.7: Confusion Matrix for proposed Model over BCI III Dataset IVa

Additionally, insufficient data may have presented a challenge, as this dataset only contains recordings from only 5 subjects, in contrast to the GigaDB dataset which had a total of 52

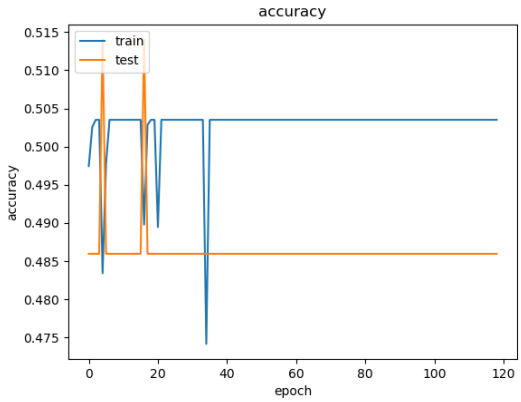
users. The case for this conjecture is reinforced by results obtained from BCI III dataset IVb, which also presented the model with foot MI as one of the classes and observed similarly poor performance, both with our model and the ResNet50.

Figures 4.8 & 4.9 representing the accuracy and loss are indicative of the model not being able to distinguish between classes and generalise. With the exception of the 2nd k-fold, where the model opted to miss-classify all of the data as belonging to the right-hand class, the training and validation accuracies and losses significantly diverge from as early as 20 epochs. While the model's training accuracy presents a continued increase, the test accuracy stagnates and breaks down at around 60%. A similar degradation is evident for the losses, with the loss reaching a low of around 0.65, after which it reverses course and increases steadily.

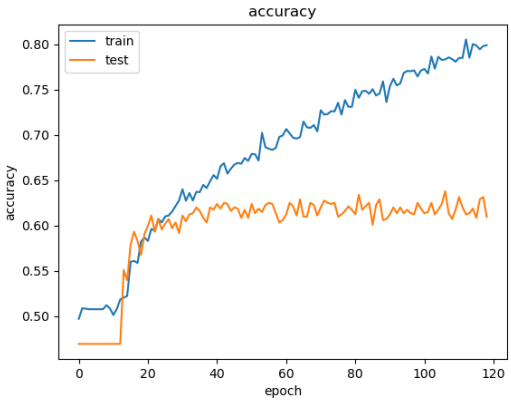
For the 2nd K-fold, the AUC is at 0.5% which indicates the model has no discrimination capacity to distinguish between the two classes. For the other k-folds, the AUC is between 0.65 and 0.70.



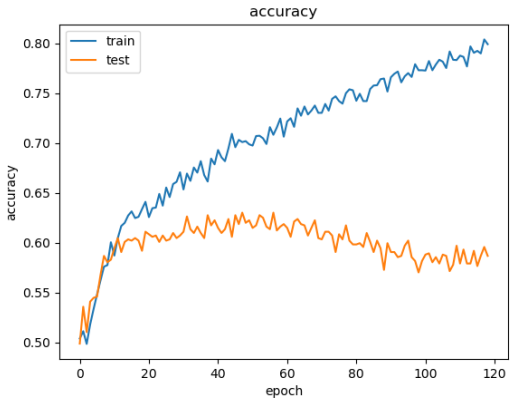
(a) K-Fold 1



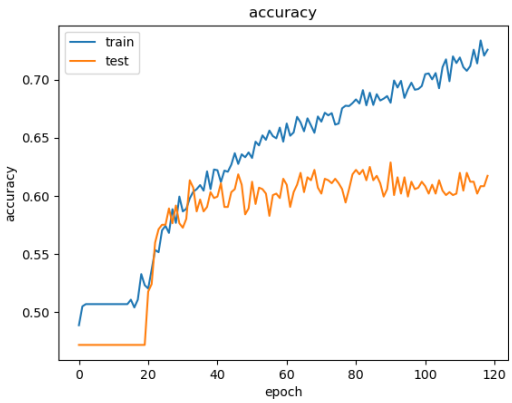
(b) K-Fold 2



(c) K-Fold 3

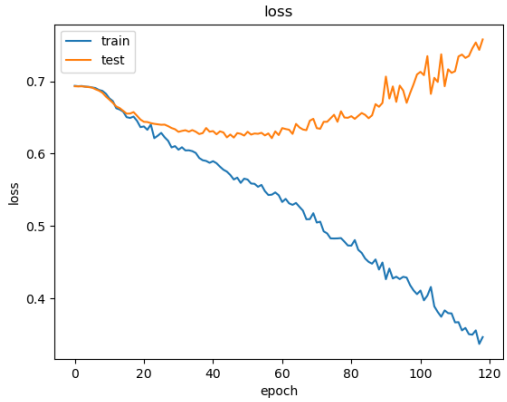


(d) K-Fold 4

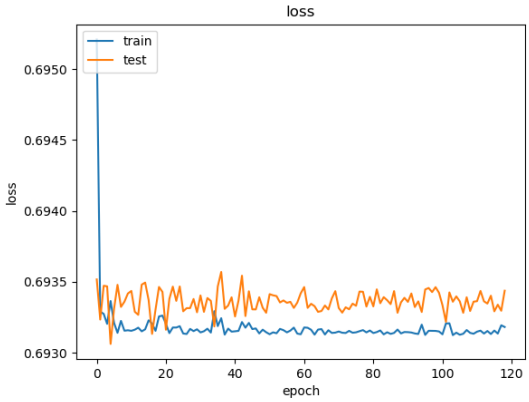


(e) K-Fold 5

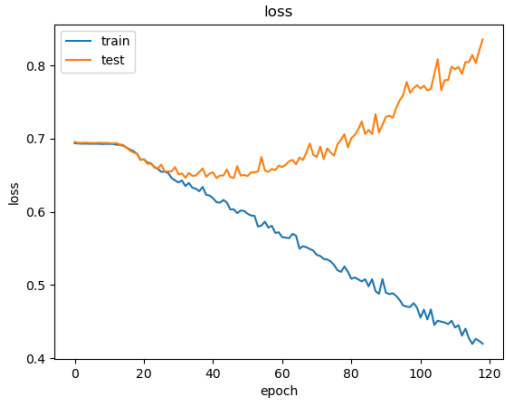
Figure 4.8: Accuracy Curves for proposed Model over BCI III IVa Dataset



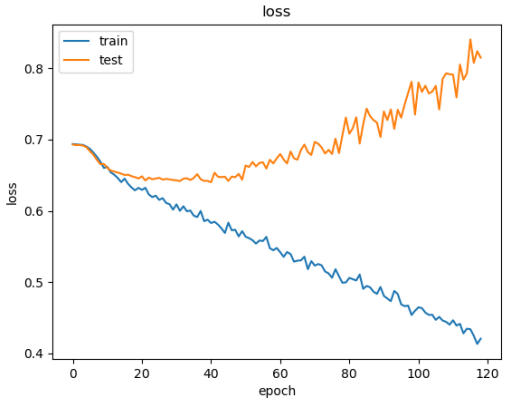
(a) K-Fold 1



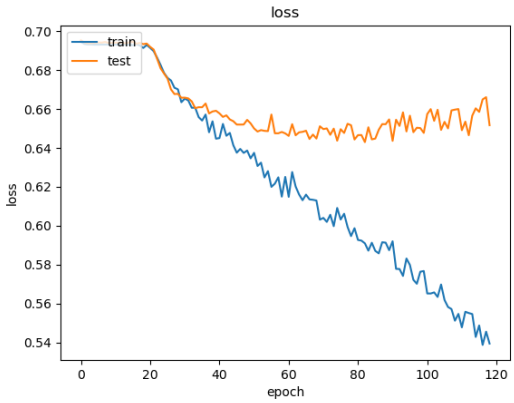
(b) K-Fold 2



(c) K-Fold 3

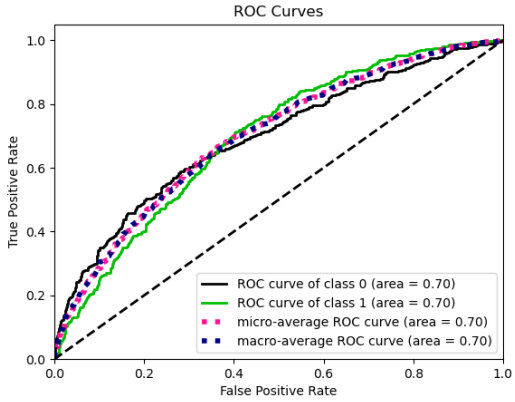


(d) K-Fold 4

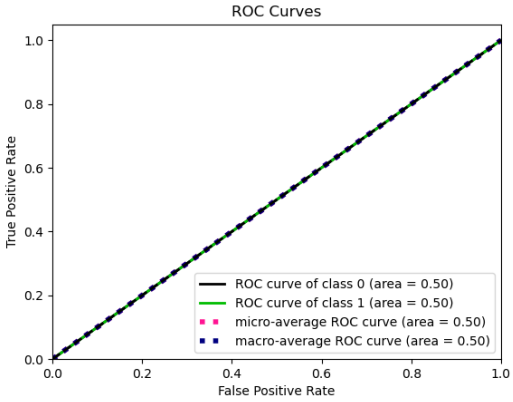


(e) K-Fold 5

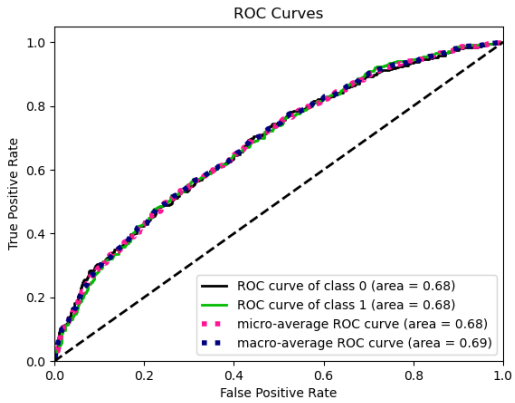
Figure 4.9: Loss Curves for proposed Model over BCI III Dataset IVa



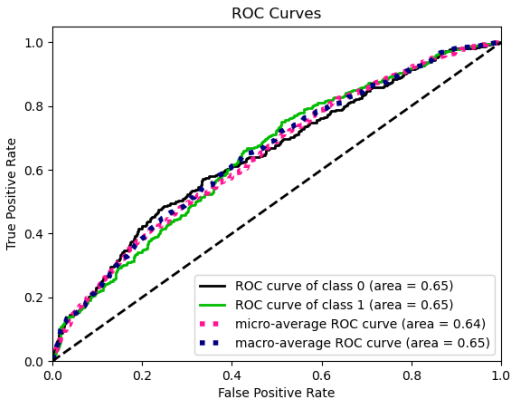
(a) K-Fold 1



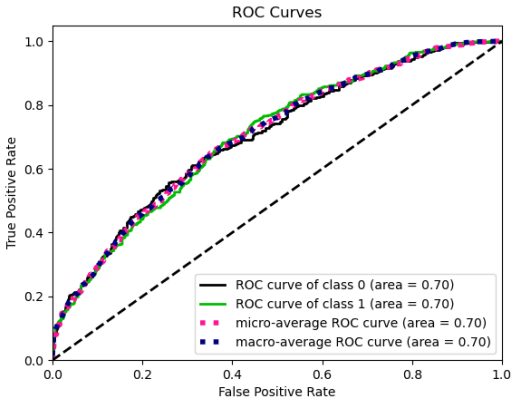
(b) K-Fold 2



(c) K-Fold 3



(d) K-Fold 4



(e) K-Fold 5

Figure 4.10: ROC Curves for proposed Model over BCI III Dataset IVa

For the IVa dataset, our network performed with an accuracy between 50% and 64.90%, with a mean of 60.10%. The poor performance was reflected in the ResNet50 network, and was even more prominent, with a mean accuracy of 54.74%. A summary of these results can be observed in Tables 4.4 & 4.5.

Table 4.4: K-Fold summary for BCI III dataset IVa

k-fold	loss	accuracy	f1	precision	recall
1	0.77	64.80	64.82	64.87	64.85
2	0.69	50.00	50.60	50.60	50.60
3	0.73	61.73	61.71	61.81	61.76
4	0.79	59.08	58.99	59.05	59.02
5	0.63	64.90	65.08	65.01	65.05
mean	0.72	60.10	60.24	60.27	60.26
std	0.06	5.49	5.31	5.31	5.31

Table 4.5: Resnet50 K-Fold summary for BCI III dataset IVa

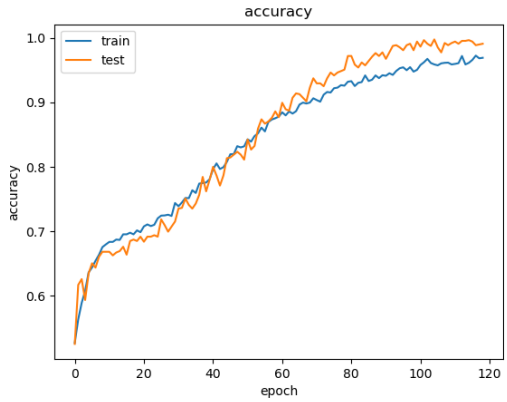
k-fold	loss	accuracy	f1	precision	recall
1	1.95	56.94	56.85	56.85	56.85
2	2.33	57.96	57.92	57.92	57.92
3	1.93	53.98	54.54	54.54	54.54
4	0.69	50.00	50.60	50.60	50.60
5	2.14	54.80	54.80	54.80	54.80
mean	1.81	54.74	54.94	54.94	54.94
std	0.58	2.77	2.51	2.51	2.51

4.5.3 BCI IV dataset I

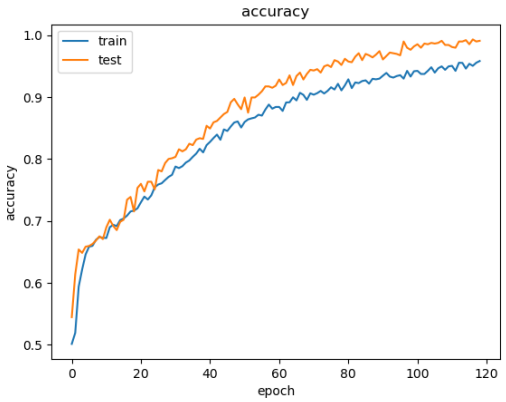
Dataset I of the BCI IV competition is another 2-class dataset with left and right hand MI EEG signals. Our model recorded its highest performance for this dataset, even outperforming ResNet50. The confusion matrices in Figure 4.11 over the 5 k-folds provide a summary of this. The model managed to appropriately classify both left and right classes, with only two instances in which the misclassification exceeded 0.01 of the total labels in the class. The 1% misclassification occurred for k-fold 4 and 5 in Sub-figures 4.11d and 4.11e.

The statistics demonstrate a disparity between the accuracy and loss scores obtained during testing and training, with testing accuracy being higher and testing loss being lower, respectively.

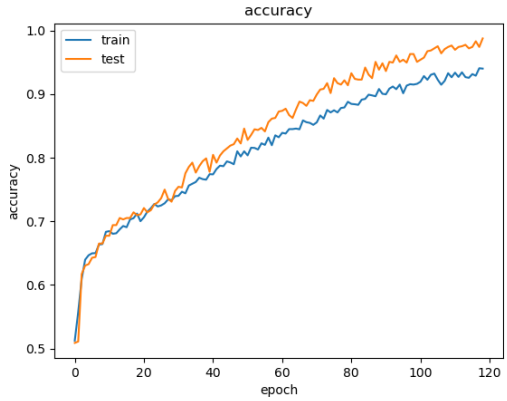
Two modes present in a deep learning model: training and testing. At testing time, regularisation procedures such as Dropout and L1/L2 weight regularisation are disabled. They are reflected in the loss of training time but not in the loss of test time. Additionally, the dCNN presents the training loss as the average of the losses for each batch of training data for the current epoch. Due to the fact that the model evolves with time, the loss over the early batches of an epoch is typically greater than the loss over the last batches. This can result in a decrease in the epoch-wise average. On the other hand, the testing loss for an epoch is computed using the model in its final state, resulting in a smaller loss.



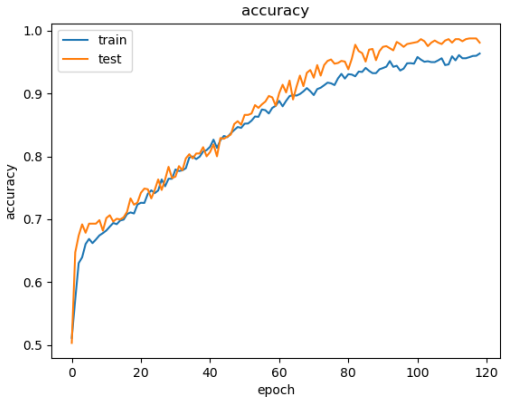
(a) K-Fold 1



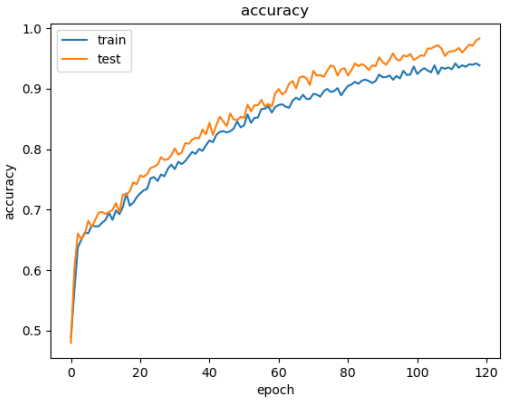
(b) K-Fold 2



(c) K-Fold 3

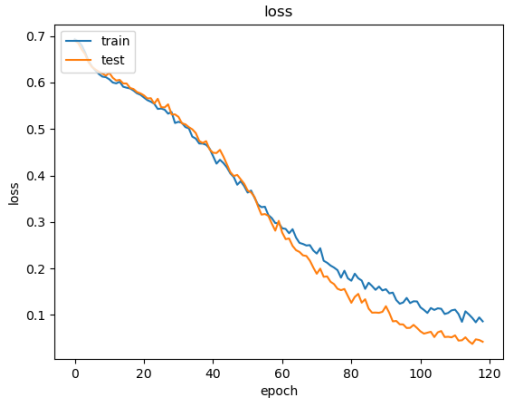


(d) K-Fold 4

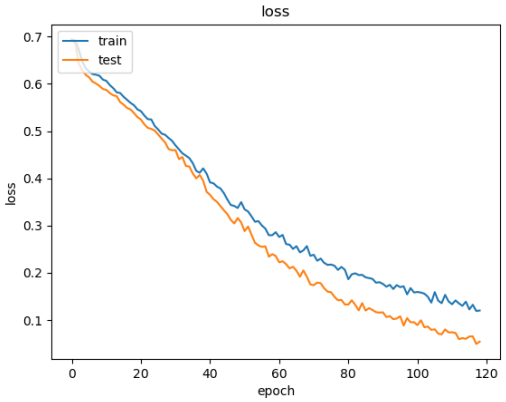


(e) K-Fold 5

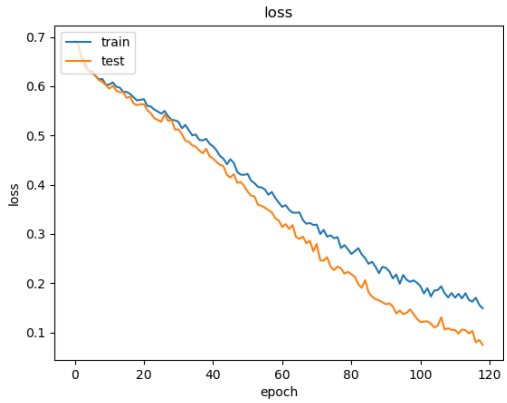
Figure 4.12: Accuracy Curves for proposed Model over BCI IV Dataset I



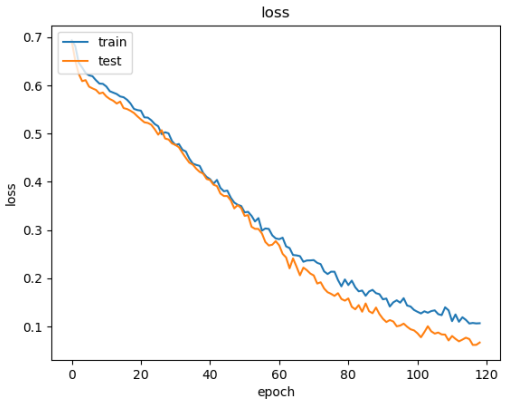
(a) K-Fold 1



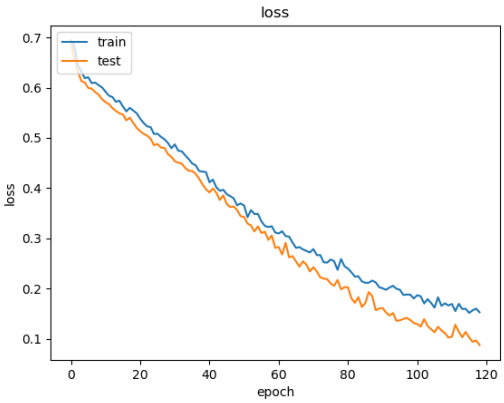
(b) K-Fold 2



(c) K-Fold 3



(d) K-Fold 4

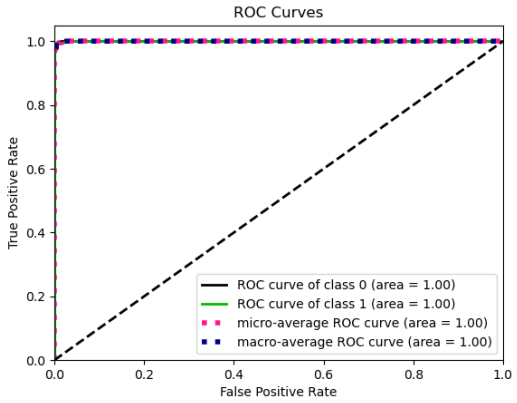


(e) K-Fold 5

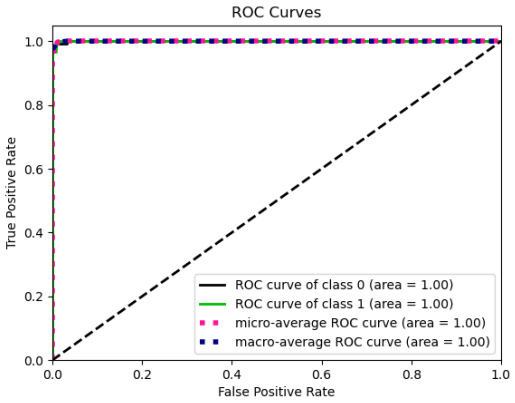
Figure 4.13: Loss Curves for proposed Model over BCI IV Dataset I

The ROC curves for our model across the dataset are depicted in Figure 4.14. The model is well adept at making correct predictions, having AUCs of 1 for all 5 k-folds. This is true for

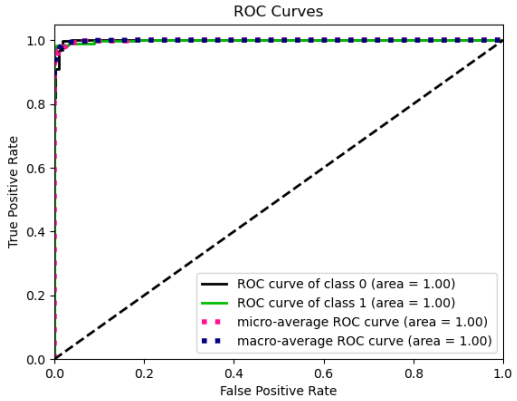
both left and right hand classes.



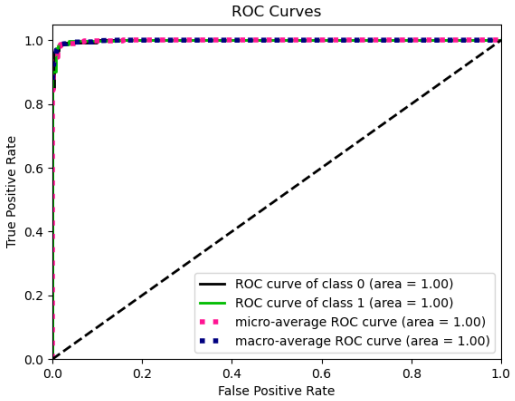
(a) K-Fold 1



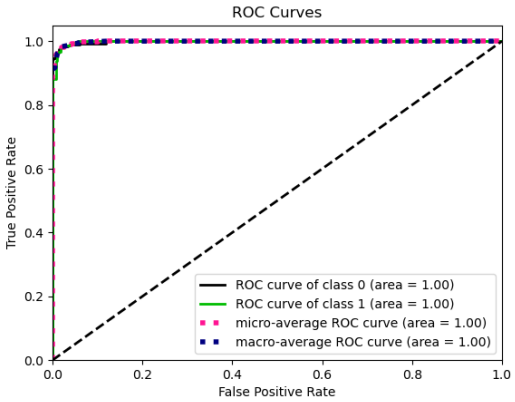
(b) K-Fold 2



(c) K-Fold 3



(d) K-Fold 4



(e) K-Fold 5

Figure 4.14: ROC Curves for proposed Model over BCI IV Dataset I

We can see by Table 4.6 that all 5 k-folds which were evaluated performed with a mean

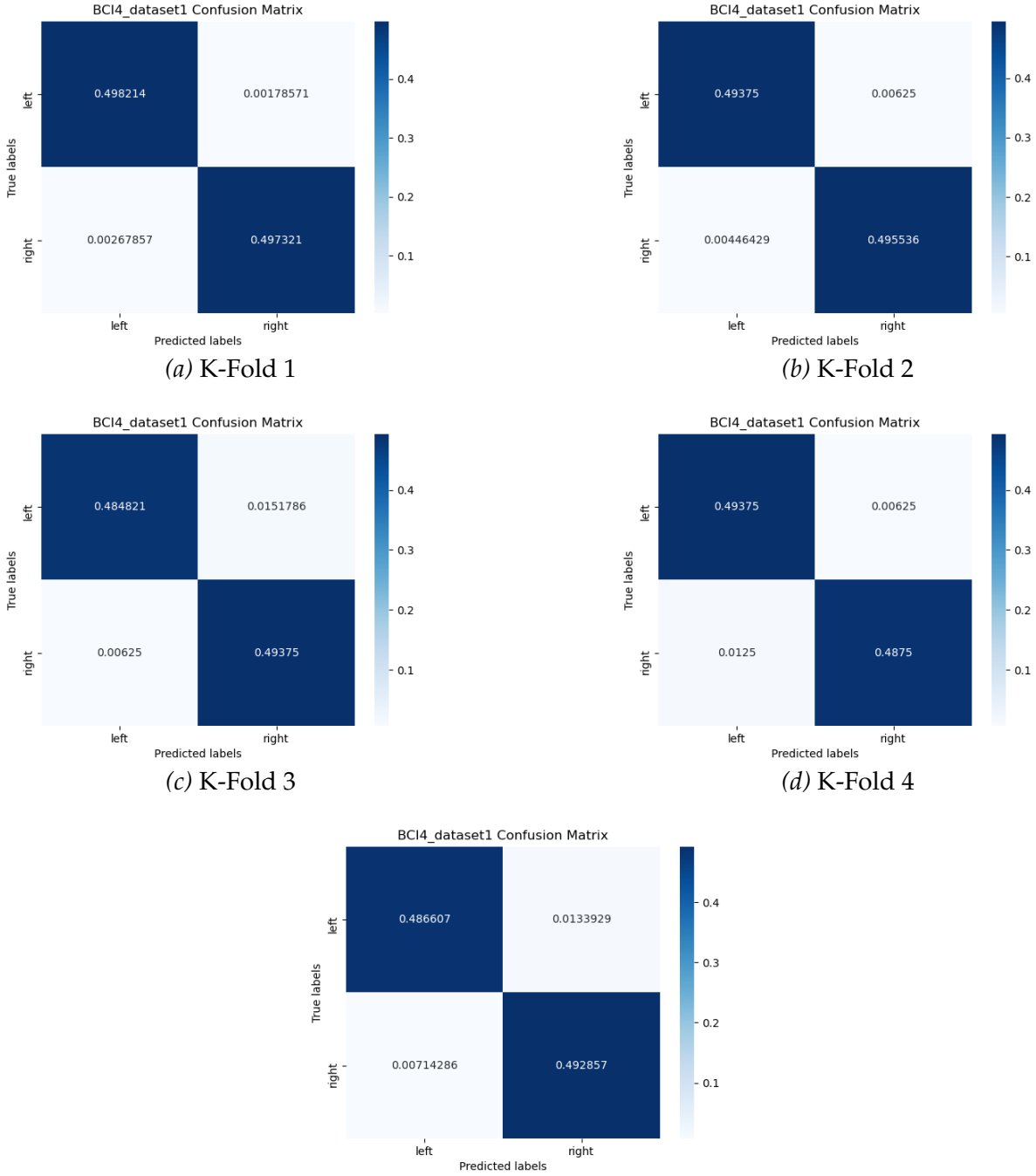
accuracy of 98.48%. The highest accuracy obtained by the model was on k-fold 1, at 99.55% with the lowest one being 97.86% for k-fold 3. Our network outperformed ResNet50 by 7.02% accuracy, on average, and had an order of magnitude lower standard deviation of 0.65 versus 6.16 for ResNet. The model's loss was also an order of magnitude lower than that of Resnet (Tables 4.6 & 4.7).

Table 4.6: K-Fold summary for BCI IV dataset I

k-fold	loss	accuracy	f1	precision	recall
1	0.04	99.55	99.55	99.55	99.55
2	0.06	98.93	98.93	98.93	98.93
3	0.09	97.86	97.77	97.85	97.81
4	0.07	98.12	98.12	98.12	98.12
5	0.08	97.95	97.95	97.95	97.95
mean	0.07	98.48	98.46	98.48	98.47
std	0.02	0.65	0.67	0.66	0.66

Table 4.7: Resnet50 K-Fold summary for BCI IV dataset I

k-fold	loss	accuracy	f1	precision	recall
1	0.09	97.14	97.14	97.14	97.14
2	0.58	86.43	86.43	86.43	86.43
3	0.05	98.84	98.84	98.84	98.84
4	0.37	92.23	92.23	92.23	92.23
5	0.87	82.68	82.68	82.68	82.68
mean	0.39	91.46	91.46	91.46	91.46
std	0.31	6.16	6.16	6.16	6.16



(a) K-Fold 1

(b) K-Fold 2

(c) K-Fold 3

(d) K-Fold 4

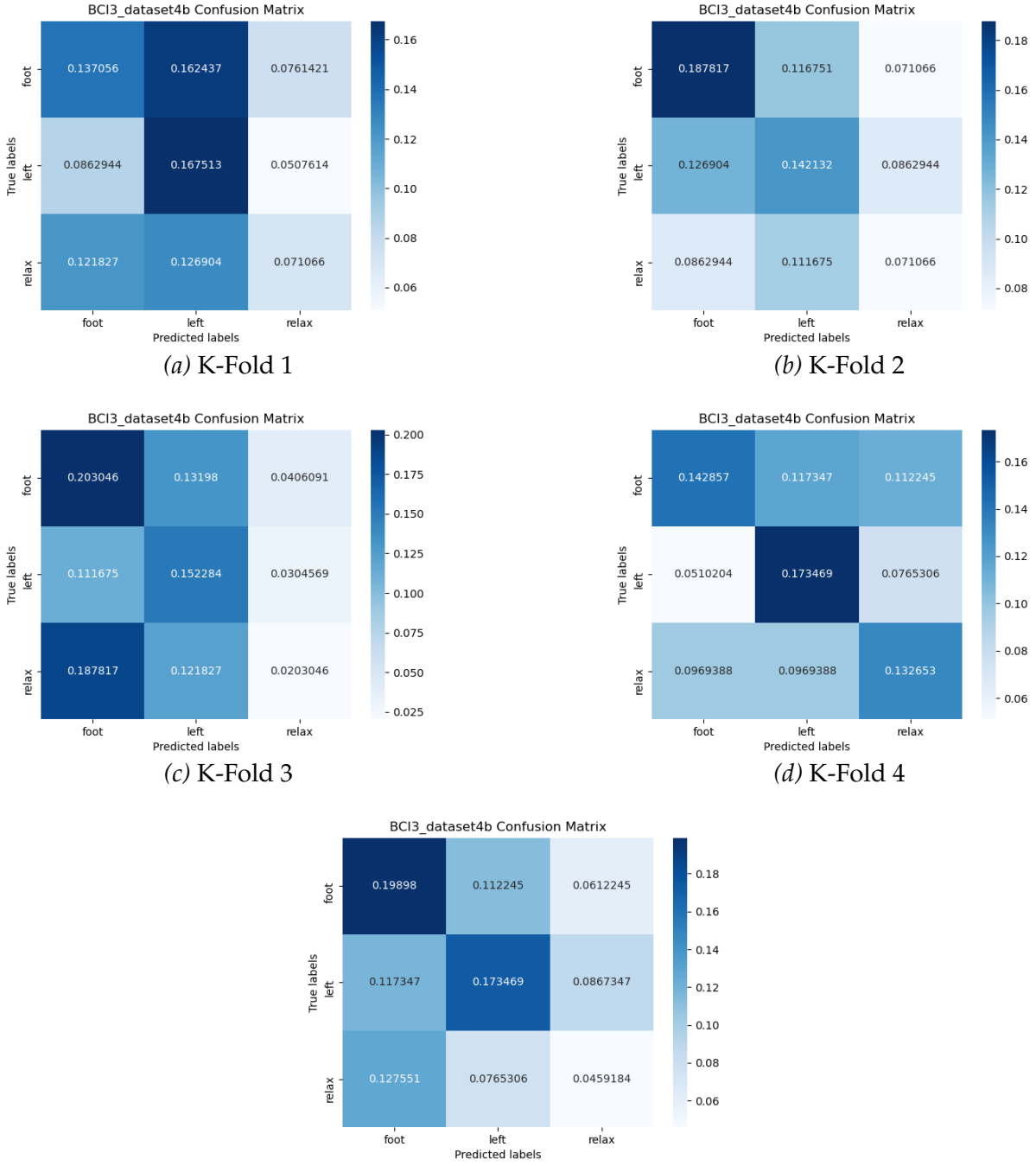
(e) K-Fold 5

Figure 4.11: Confusion Matrix for proposed Model over BCI IV Dataset I

4.5.4 BCI III dataset IVB

Dataset BCI III IVb is a 3 class dataset which contains data from a single individual. The classes in question are *right hand*, *foot* and a *relax* state. Though it was initially recorded with 2 classes, a 3rd rest class was labelled within the data. Dataset BCI III IVa presented a real challenge for both our network as well as for ResNet50. In general, human EEG signals vary according to each individual's mental state, resulting in a high degree of inter-subject variability. While existing ML and DL algorithms have demonstrated progress in MI classification, many suffer from lack of data on new users. As will be made apparent in section 4.5.5, the model does not have any particular shortcomings in handling 3 class problems. Therefore, the performance on the dataset in question may be linked to its peculiarities. It is, as such, worthy of discussion.

The confusion matrices in Figure 4.11d can help in understanding the poor performance of our model on this dataset. As can be observed, the model is not able to generalise for any of the k-folds. However, pattern is observable. In 4 out of the 5 k-folds, it heavily misclassified the *relax* class as either *right hand* or *foot*. This would indicate that as observed in Section 4.5.2, the model struggled even with a 2 class problem in the case where one of the classes was foot MI. The conjecture, in that instance, was that foot EEG signals subside more rapidly and are, thus, more difficult to distinguish from noise. Allowing for that assumption, the problem may be compounded by the introduction of a *relaxed* state, which could present with a similar noise profile. As such, foot data may be very difficult to distinguish from a relaxed state. Additionally, a relaxed state could be difficult to classify as the model attempts to look for subtle patterns which get amplified in the remaining classes.



(a) K-Fold 1

(b) K-Fold 2

(c) K-Fold 3

(d) K-Fold 4

(e) K-Fold 5

Figure 4.15: Confusion Matrix for proposed Model over BCI III Dataset IVb

As is apparent from the confusion matrices and from the ROC curves in Figure 4.16, the model has little discrimination capacity between the 3 classes, with AUCs in the range of

0.54 to 0.63.

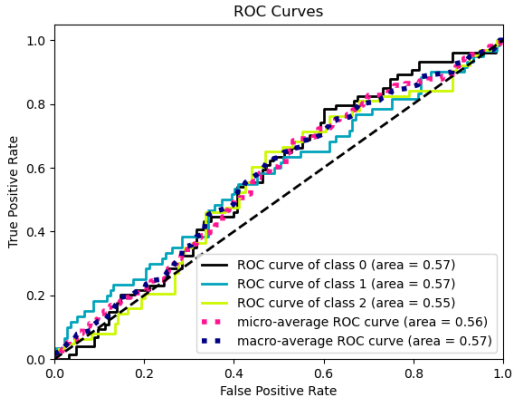
Both our model and ResNet50 performed equally poorly on the dataset (Tables 4.8 & 4.9), with a mean accuracy of 40.39% for our model and 37% for ResNet50. Our model had a higher maximum accuracy of 44.9% and a higher standard deviation of 2.78.

Table 4.8: K-Fold summary for BCI III dataset IVb

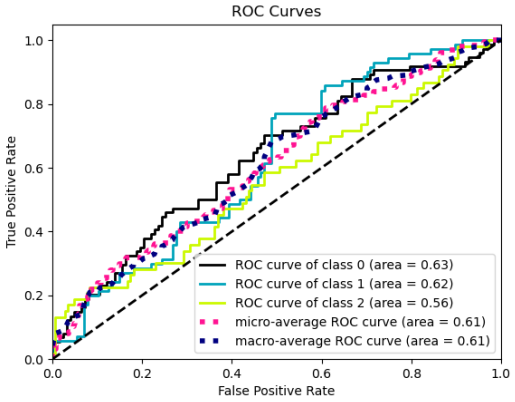
k-fold	loss	accuracy	f1	precision	recall
1	1.16	37.56	61.07	35.89	45.17
2	1.14	40.10	58.57	38.91	46.68
3	1.07	37.56	43.48	36.71	39.69
4	1.13	44.90	48.21	31.09	37.78
5	1.12	41.84	58.48	36.47	44.87
mean	1.12	40.39	53.96	35.81	42.84
std	0.03	2.78	6.86	2.57	3.46

Table 4.9: Resnet50 K-Fold summary for BCI III dataset IVb

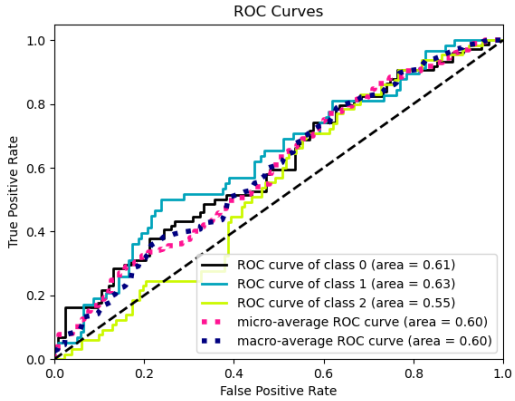
k-fold	loss	accuracy	f1	precision	recall
1	21.21	37.56	33.04	33.04	33.04
2	3.93	35.53	31.25	31.25	31.25
3	4.02	37.56	33.04	33.04	33.04
4	1.11	37.24	0.00	0.00	0.00
5	2.92	37.76	33.04	33.04	33.04
mean	6.64	37.13	26.07	26.07	26.07
std	7.36	0.82	13.06	13.06	13.06



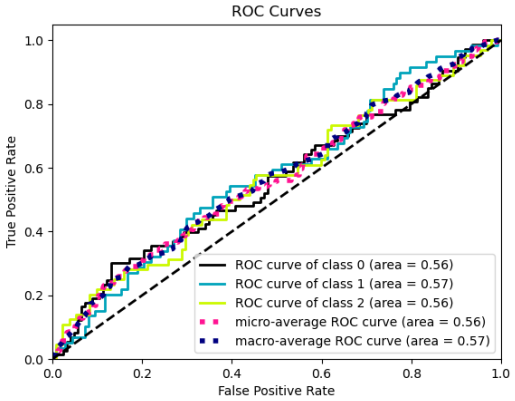
(a) K-Fold 1



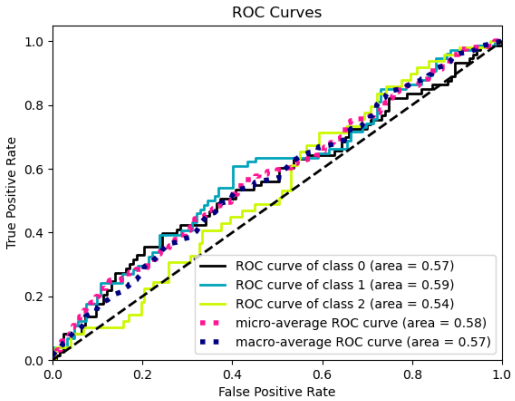
(b) K-Fold 2



(c) K-Fold 3



(d) K-Fold 4



(e) K-Fold 5

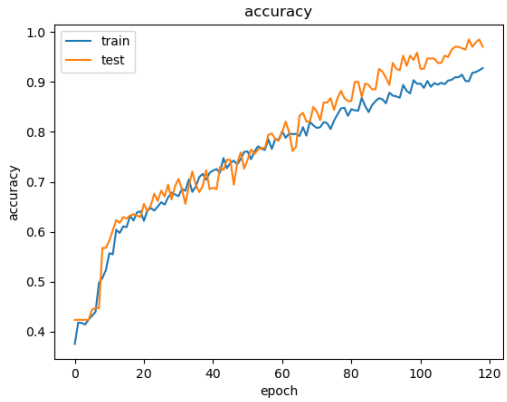
Figure 4.16: ROC Curves for proposed Model over BCI III Dataset IVb

4.5.5 BCI III dataset V

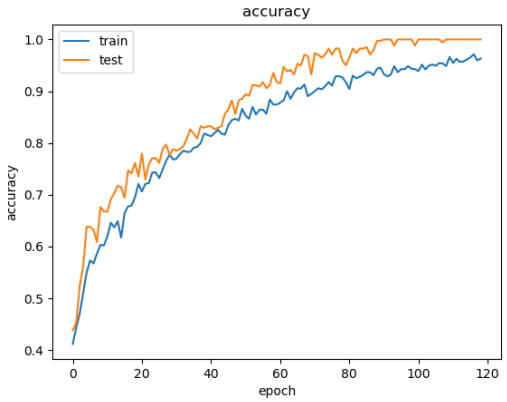
Dataset 5 of the BCI III competition presents a three class database with EEG signals collected from 3 subjects. The classes in question are *left*, *right* and *word image*. Our model recorded the highest score on this dataset.

The confusion matrices in Figure 4.17 demonstrate the model classifying the 3 classes exceptionally well, with no misclassification errors in 3 out of the 5 k-fold instances. It is to be noted that the dataset is not evenly balanced, with the 3rd class (*word image*) representing over 40% of the data points, and with a higher number of right hand recordings than left ones.

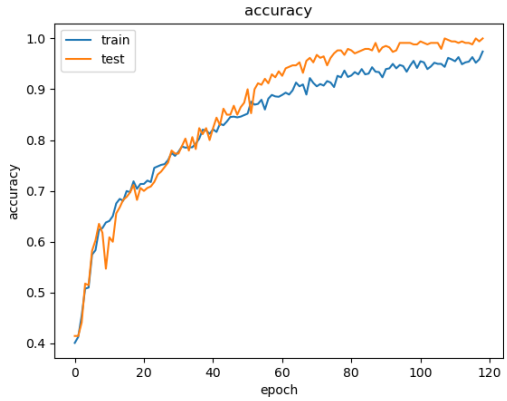
As seen in the Figure 4.18, both training and validation accuracy closely track one another, with little divergence, for all k-folds. The same is true for the loss curves in Figure 4.19. The figures indicate a discrepancy between testing and training scores for accuracy and loss, with testing accuracy being higher than during training, and testing loss being lower, respectively. As discussed in dataset Section 4.5.2, deep learning has two modes: training and testing. Dropout and L1/L2 weight regularisation are disabled during testing. They affect loss and accuracy at training time but not at test time. For instance, a Dropout of 0.25 means that 25% of features will be zero throughout training. However, all attributes are employed during testing, which makes the model more robust and improves testing accuracy. The implication is that accuracy and loss curves test metrics are higher than the ones during training, hence the gaps between the training and testing graphs.



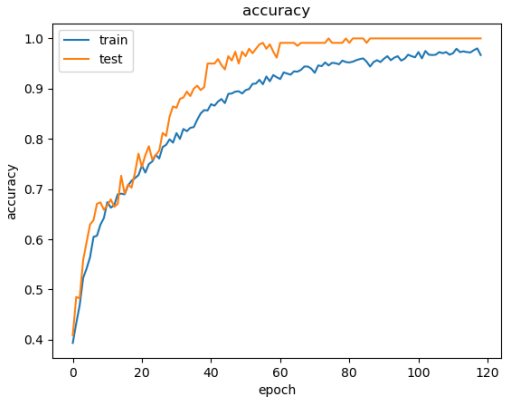
(a) K-Fold 1



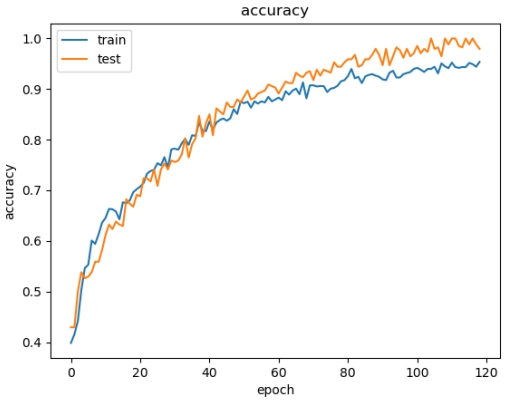
(b) K-Fold 2



(c) K-Fold 3

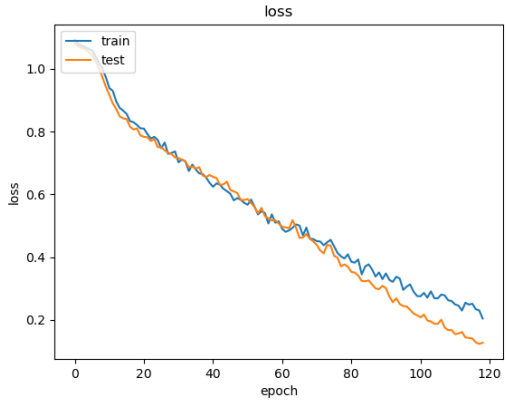


(d) K-Fold 4

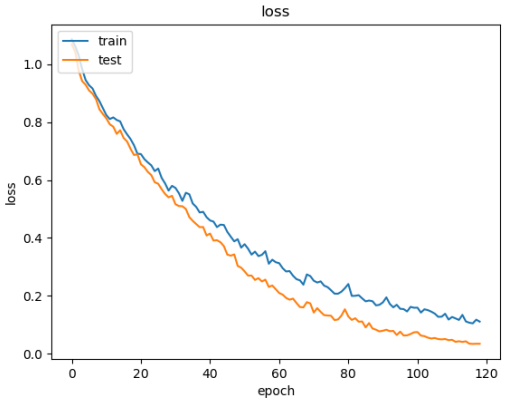


(e) K-Fold 5

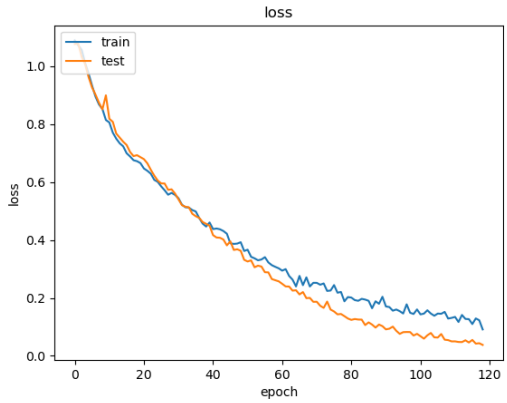
Figure 4.18: Accuracy Curves for proposed Model over BCI III Dataset V



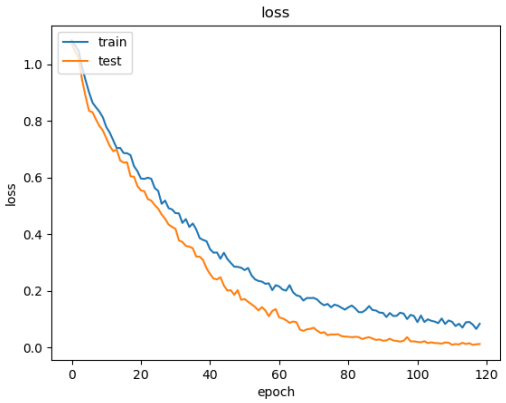
(a) K-Fold 1



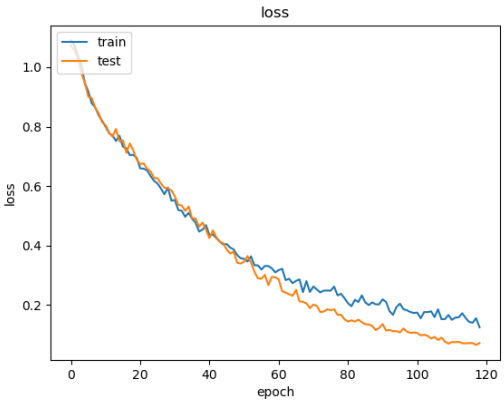
(b) K-Fold 2



(c) K-Fold 3



(d) K-Fold 4

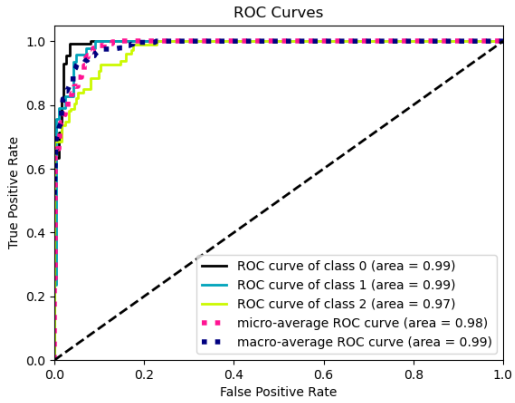


(e) K-Fold 5

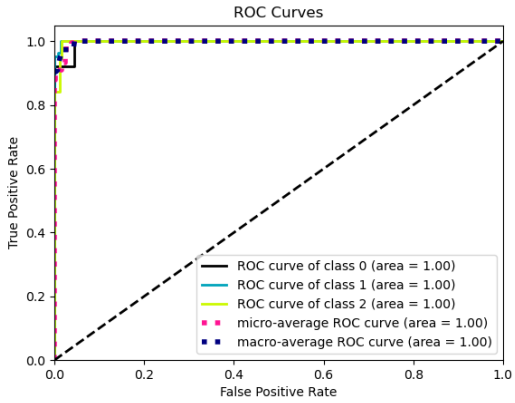
Figure 4.19: Loss Curves for proposed Model over BCI III Dataset V

Our model's ROC curves are depicted in Figure 4.20, which shows the curves for our model across the whole dataset. The model is adept at making correct predictions, with AUCs

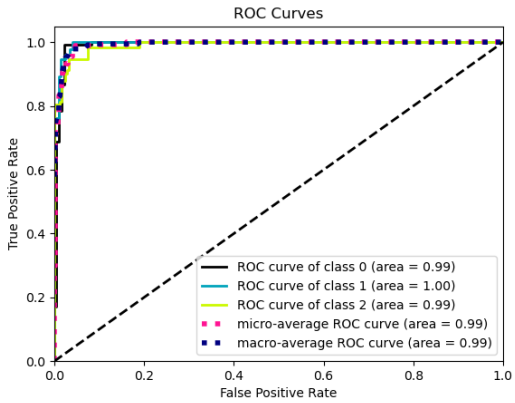
between 1 and 0.97 for all 5 k-folds in the data set. This is true for all 3 classes, *left*, *right* and *word image*.



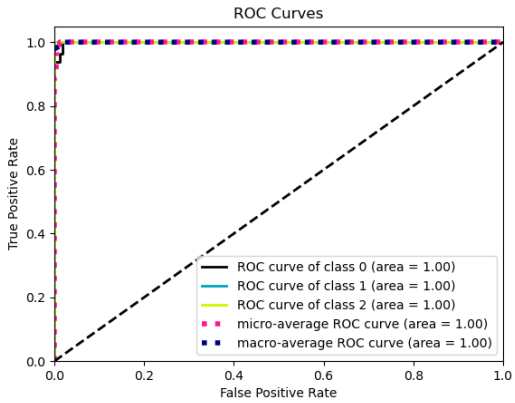
(a) K-Fold 1



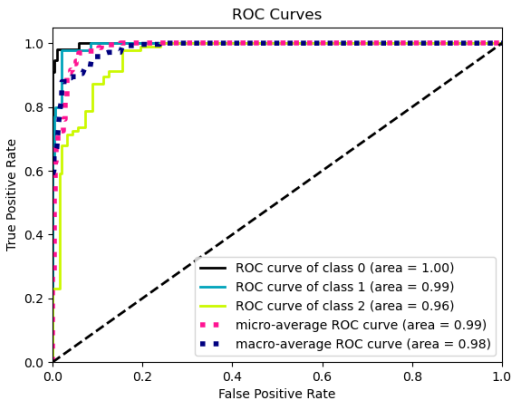
(b) K-Fold 2



(c) K-Fold 3



(d) K-Fold 4



(e) K-Fold 5

Figure 4.20: ROC Curves for proposed Model over BCI III Dataset V

A summary of the model's performance metrics is presented in Table 4.6. The accuracy of our proposed model ranged between 98.82% and 100% with a standard deviation of 0.49 and a mean loss of 0.06.

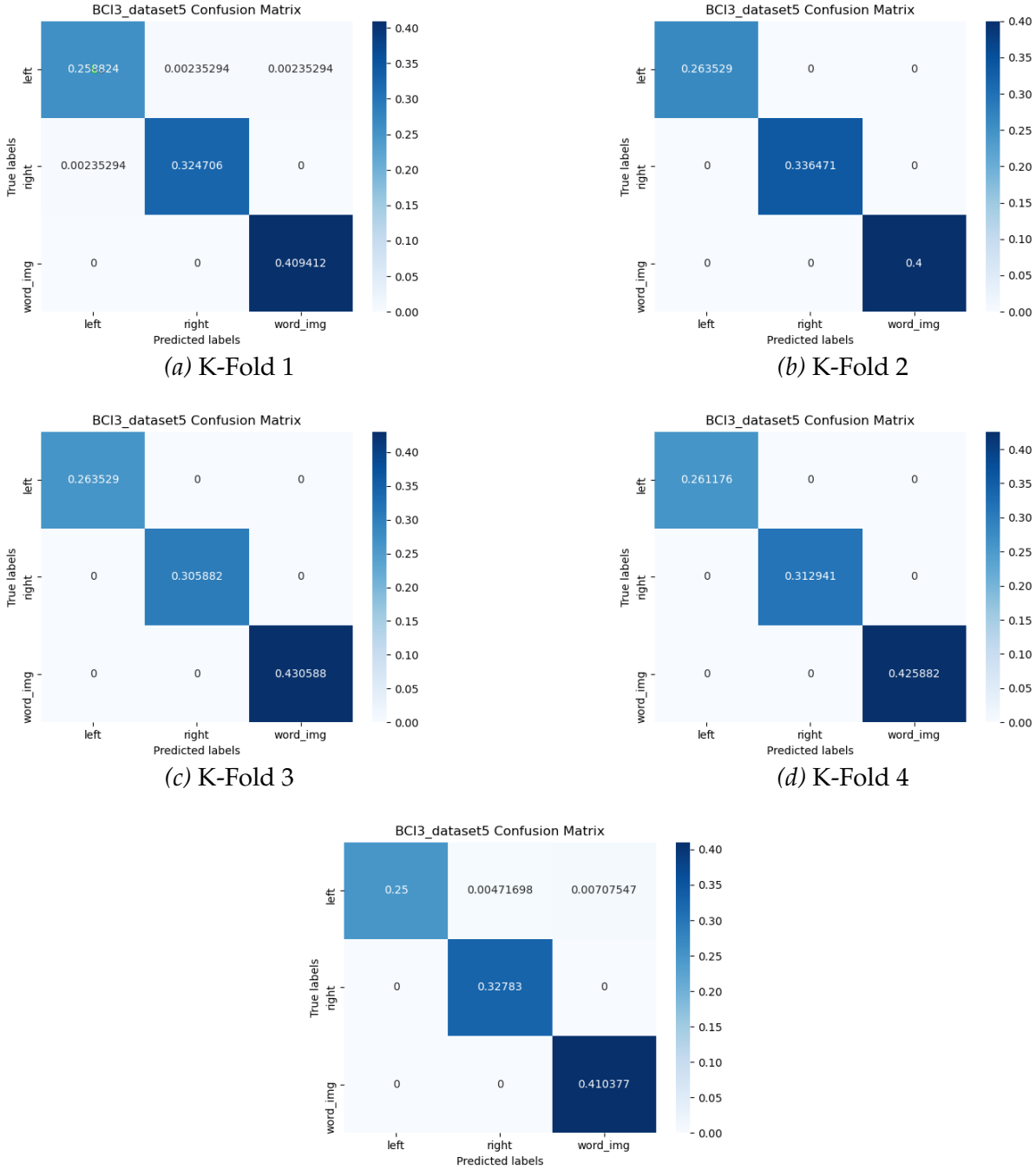
Table 4.10: K-Fold summary for dataset BCI III Dataset V

k-fold	loss	accuracy	f1	precision	recall
1	0.11	99.29	99.29	99.29	99.29
2	0.04	100.00	100.0	100.0	100.0
3	0.05	100.00	100.0	100.0	100.0
4	0.02	100.00	100.0	100.0	100.0
5	0.06	98.82	98.82	98.82	98.82
mean	0.06	99.62	99.62	99.62	99.62
std	0.03	0.49	0.49	0.49	0.49

Comparatively, ResNet50 did not perform well, as it did not manage to generalise for the 3-class problem (Table 4.11). It obtained a mean accuracy of 34.93%.

Table 4.11: Resnet50 K-Fold summary for dataset BCI III Dataset V

k-fold	loss	accuracy	f1	precision	recall
1	70.00	32.71	31.03	31.03	31.03
2	6.59	40.00	43.08	43.08	43.08
3	51.41	43.06	45.98	45.98	45.98
4	3.31	26.12	24.78	24.78	24.78
5	61.49	32.78	31.03	31.03	31.03
mean	38.56	34.93	35.18	35.18	35.18
std	28.09	5.98	8.02	8.02	8.02



(a) K-Fold 1 *(b)* K-Fold 2

(c) K-Fold 3 *(d)* K-Fold 4

(e) K-Fold 5

Figure 4.17: Confusion Matrix for proposed Model over BCI III Dataset V

4.5.6 BCI III dataset IIIA

A final difficult challenge for the model was BCI III Dataset IIIa, a dataset with 4 classes, namely *feet*, *left*, *right* and *tongue*. The data was collected from 3 subjects.

As can be observed from the confusion matrices in Figure 4.21, the model had a very difficult time distinguishing between the 4 classes of data.

As discussed in a previous section (Section 4.5.4), one possible explanation for this behaviour is that the foot EEG signals may present as challenging to distinguish from noise, given its cortical topology. When compared to the motor cortex responsible for foot movement, the motor cortex responsible for hand movement has a significantly larger surface area. Insufficient training data could also account for the poor performance of the network, as the EEG signals originate from only 3 users. However, the confusion matrix (Figure 4.21) and the ROC curves (Figure 4.24) seem to suggest that, over all of the k-folds, the model struggled more in distinguishing left-hand signals.

Given the higher number of classes, the low number of subjects and the difficulties encountered by both our proposed network and ResNet50 in classifying feet data, further exploration is required. One avenue to explore would be to remove the feet data from the dataset and evaluate the model's performance on a 3-class problem, similar to Section 4.5.5, which yielded very promising results. Further research is required to address the difficulty of classifiers to handle feet data. Investigations into alternative solutions may shed light on the cause of the problem. To address the problem of insufficient data, the combination of multiple databases in the training of future models could offer possible improvements. Additionally, along with the growing interest in the fields of BCI using DL, the authors expect future databases to become available and provide ample data for research.

(a) K-Fold 1

(b) K-Fold 2

(c) K-Fold 3

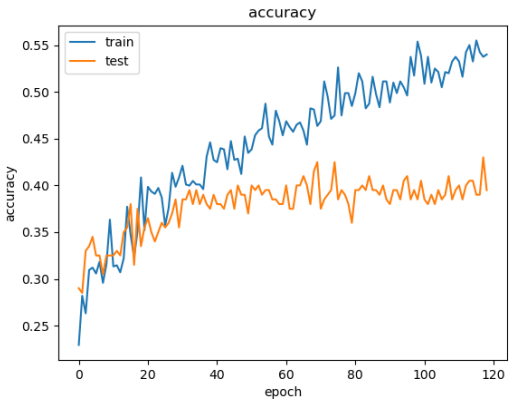
(d) K-Fold 4

(e) K-Fold 5

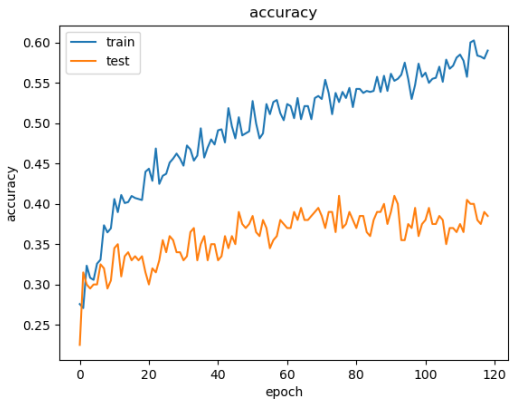
Figure 4.21: Confusion Matrix for proposed Model over BCI III Dataset IIIa

Figures 4.22 and 4.23 indicate a very wide over-fitting gap emerging very early on in the training phase, with the training metrics continuing to improve but the test metrics stagnat-

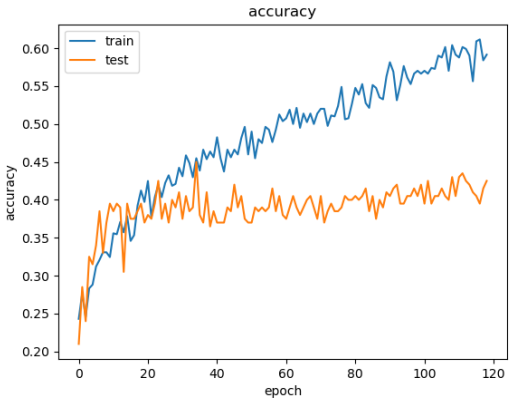
ing or reversing.



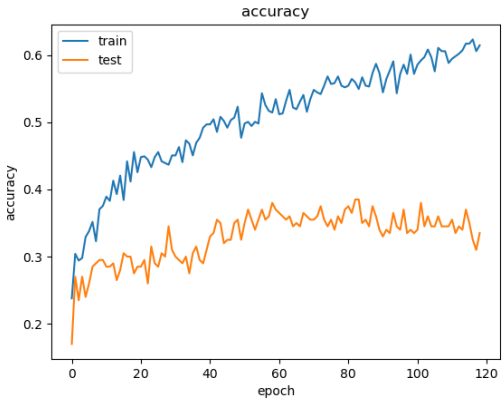
(a) K-Fold 1



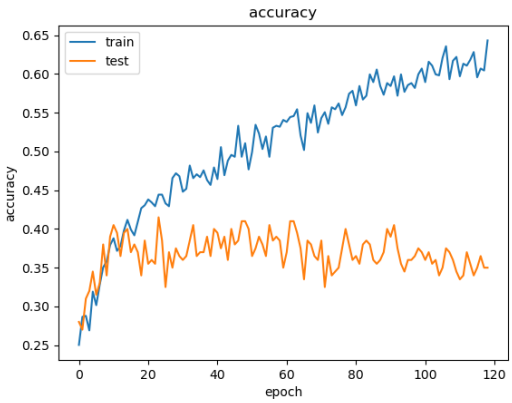
(b) K-Fold 2



(c) K-Fold 3



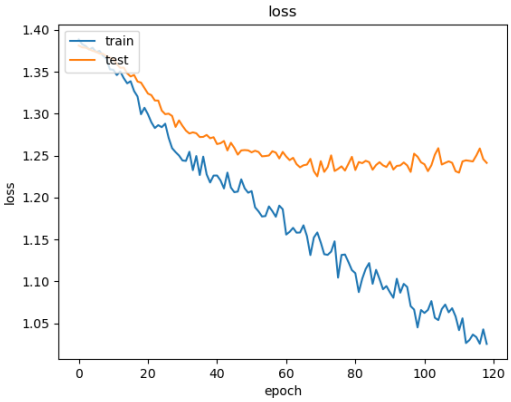
(d) K-Fold 4



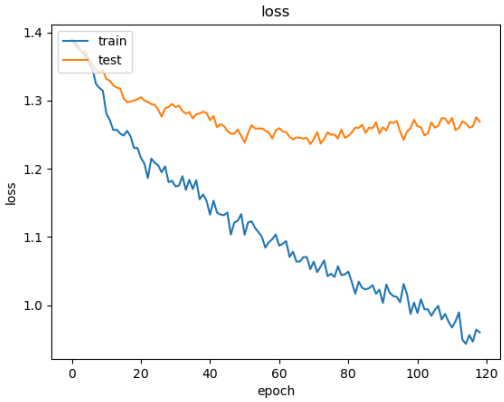
(e) K-Fold 5

Figure 4.22: Accuracy Curves for proposed Model over BCI III Dataset IIIa

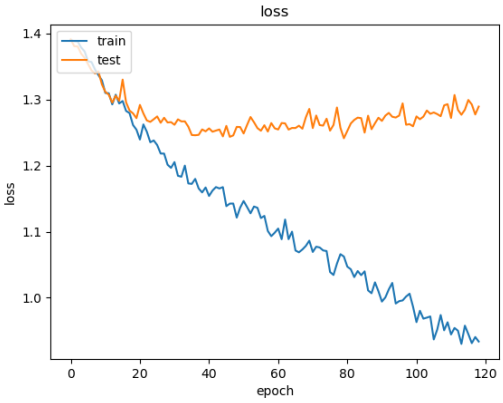
According to the AUCs in Figure 4.24, the model is hardly able of distinguishing between



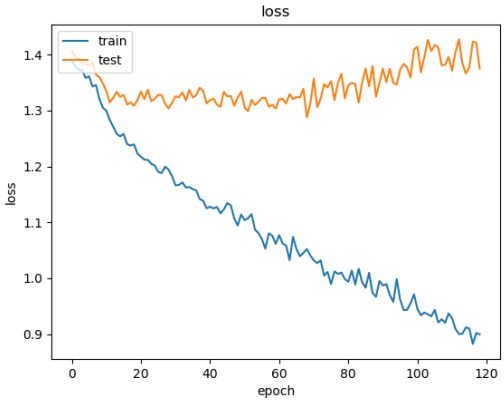
(a) K-Fold 1



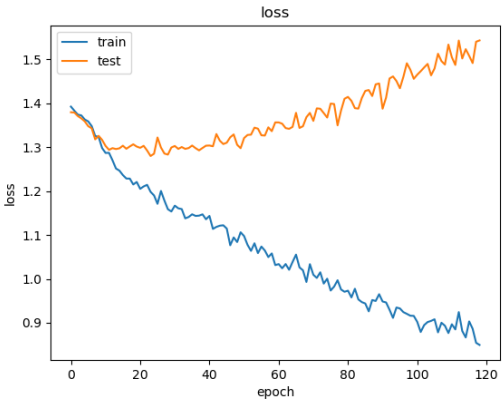
(b) K-Fold 2



(c) K-Fold 3



(d) K-Fold 4



(e) K-Fold 5

Figure 4.23: Loss Curves for proposed Model over BCI III Dataset IIIa

the 4 classes. The AUCs over the 5 k-folds are in the range of 0.62 to 0.75.

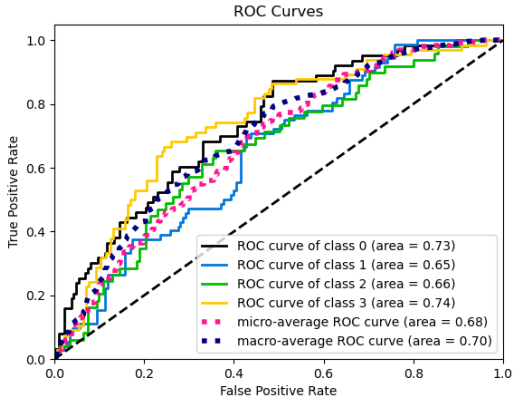
Table 4.12: K-Fold summary for BCI III Dataset IIIa

k-fold	loss	accuracy	f1	precision	recall
1	1.30	38.00	38.91	33.54	46.34
2	1.33	41.60	37.70	32.45	44.99
3	1.28	47.20	39.56	34.16	47.01
4	1.33	41.77	38.70	33.61	45.61
5	1.32	39.76	39.76	34.53	46.88
mean	1.31	41.67	38.93	33.66	46.17
std	0.02	3.09	0.72	0.71	0.77

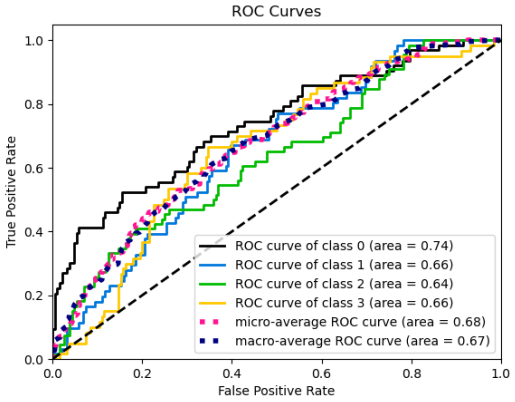
Table 4.13: Resnet50 K-Fold summary for BCI III Dataset IIIa

k-fold	loss	accuracy	f1	precision	recall
1	2.09	28.80	28.12	28.12	28.12
2	3.07	26.40	25.78	25.78	25.78
3	1.57	28.80	5.47	31.25	9.17
4	2.57	24.90	24.22	24.22	24.22
5	4.44	30.92	32.81	32.81	32.81
mean	2.75	27.96	23.28	28.44	24.02
std	0.98	2.10	9.37	3.22	7.97

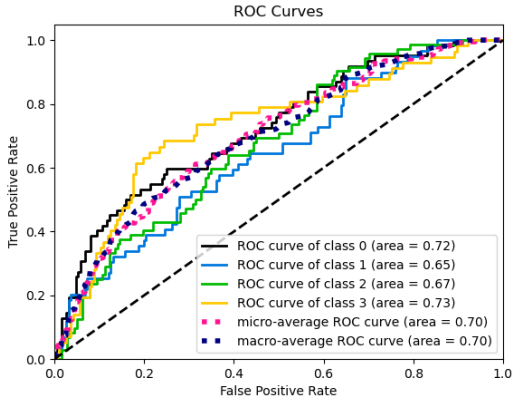
A summary of these performance metrics for our model and for ResNet50 can be found in Tables 4.12 & 4.13. While both networks performed very poorly, the performance of ResNet was twice as bad, on average. Its mean accuracy was 27.96%, while our network's mean was at 41.76%. The accuracy of our network was between 38.00% and 41.67%, while the highest accuracy achieved by ResNet was 28.80%.



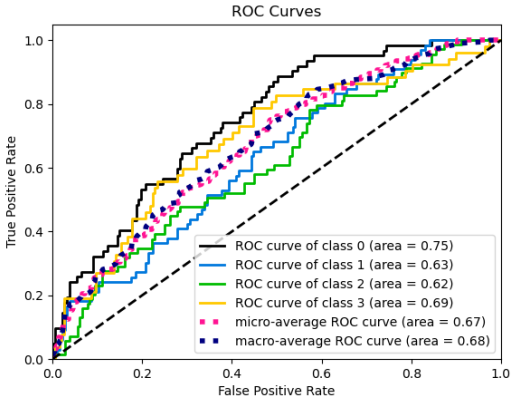
(a) K-Fold 1



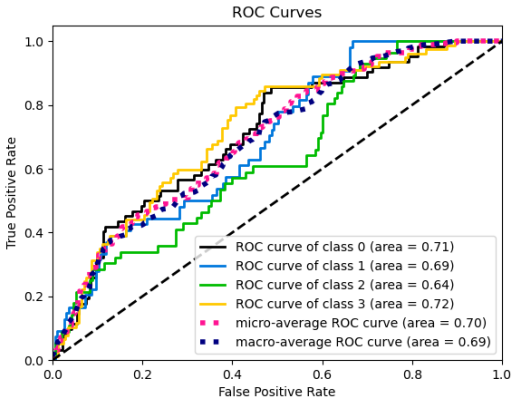
(b) K-Fold 2



(c) K-Fold 3



(d) K-Fold 4



(e) K-Fold 5

Figure 4.24: ROC Curves for proposed Model over BCI III Dataset IIIa

4.6 Conclusions

In this work, a comprehensive study of the use of topographical representations of EEG as a preprocessing step combined with a deep convolutional network were explored.

The proposed framework and model has a number of characteristics to address the drawbacks associated with conventional ML methods. The ability of the proposed CNN model to learn its own features from the transformed data provides a significant advantage for possible real-life applications. Handcrafting features from raw data or even the transformed topographical maps by human experts leaves much room for error and sometimes this can involve the extraction of features that are not related to the classes involved, thus effecting the model's performance.

On average, the proposed framework in this chapter matched performance of state-of-the-art algorithms for binary class datasets specifically for right-left MI. It was observed that the overall dataset exhibits a consistent pattern of high-accuracy performance across multiple metric parameters, as indicated in the results discussion. By pre-processing the data prior to passing it to the CNN, it has been demonstrated that the deep network is capable of capturing the relevant patterns within the data.

Due to the small size of the networks used and the reduced number of parameters to learn, it makes it more feasible to be embedded into smart assistive devices. The use of small CNN models reduces training costs while maintaining an acceptable response time. However, as the number of classification categories increased, i.e. three-class and four-class, performance decreased.

The framework and model were optimised for the H.Cho database (i.e. GigaDB [159]). The H.Cho dataset is a two-class dataset comprised of images of the right and left hands, which has the most participants (52), and hence a wider pool of training data. While reviewing the data, it was observed that the algorithm is capable of successfully classifying left and right hand MI with a high degree of classification accuracy. However, when the framework was

tested against a binary framework, such as right hand and foot for BCI III dataset IVa, we can see that the data is not being evaluated appropriately. This could be down to a number of reasons, such as the number of data points, need to adjust the model such that there are more layers to find different relevant details within the data. However, in BCI III dataset V, we can see that we can correctly categorise using left and right hand imagery, and the system was able to infer that the remaining data belonged to the word-img category. It can be concluded that the proposed network is capable of performing robustly across binary left right MI datasets.

The observations from our experiments suggest that augmenting EEG data while retaining its temporal and spatial information can be beneficial to the performance of a dCNN. According to the data presented in the preceding sections, the model achieved a highly acceptable and dependable level of accuracy. Regularisation supported the model's resilience against overfitting. This was accomplished through the use of dropout and pooling layers, k-fold cross validation, and batch learning procedures.

This framework outperforms other frameworks in tasks involving MI classification. The results corroborate previously published accuracy values in related studies as well as our own comparative experiments against ResNet. It is not commonly seen in literature for EEG-MI classification to find a model looking for overall accuracy of a model. This is not the case for the proposed framework and model as it can be deduced that it is able to holistically look at the overall data using temporal, spatial and spectral information derived from the topographical maps. Not only will training ML and DL models on appropriate features improve their performance and robustness, but they will also lower computational costs associated with model training and deployment for real-life applications.

In practice, training a specialised network for a single subject can be time consuming, since it demands not only a large amount of time but also a large amount of computer memory. Thus, in this research all subjects' training data to create a single neural network capable of learning the common features of all subjects.

The following factors may contribute to the shared neural network's superiority over separate DNNs: To begin, the combined dataset is far larger than the individual datasets. A larger training set that includes not only the subject's MI-EEG features but also the MI features of other subjects can increase deep neural network classification accuracy. Second, we minimised DNN's overfitting. The integrated data set includes both common and unique characteristics of MI tasks. Individual variations can cause neural networks to experience less overfitting and hence learn more common features.

The application of DL approaches to EEG processing is promising and warrants more attention and further study. The suggested framework can be adapted for BCIs and has the potential to be used in medical solutions for rehabilitation and regaining autonomy by interfacing with EEG-compatible smart devices, and these features can extend beyond the medical, into other fields.

5

Concluding remarks and recommendations for Future work

5.1 Discussion

For nearly two decades, BCI systems have been extensively studied in order to rehabilitate individuals with motor impairments. The researchers want to make this technology accessible to everyone, despite the fact that it is prohibitively expensive for end users. BCI has developed into an intriguing research topic and a critical subject for a variety of applications. It is a way for a computer and a brain to communicate in order to perform an external procedure. BCI aids the current workings of the nerves and muscles and generates motor tasks using EEG signals, which are then translated into physical behaviour using hardware and software.

The purpose of this work was to improve the accuracy of detection frameworks using DL frameworks, thereby allowing for a more precise definition of user intention based on EEG data. This thesis presented two frameworks for classifying mental tasks. To begin, we build

a power spectrum image using spectrograms, which is utilised as an input to a dCNN model. We explored the accuracy that can be achieved when a two-class, and four-class model are employed.

There were two proposed frameworks based on dCNNs to perform feature extraction and classification for MI EEG signals. The results demonstrated that when compared to other conventional ML techniques and state of the art models, the dCNN can improve classification performance. The studies demonstrate that the proposed framework with topographical maps is capable of robustly and effectively classifying MI for two- and three-class datasets, and provides a practical framework for BCI applications using non-invasive EEG signals.

These were the questions posed at the beginning of this thesis (Section 1.2) with regards to the spectrogram-dCNN chapter. Firstly, would providing a spectrogram, frequency, representation of EEG signals to a DNN improve on results obtained from classical ML and state–of–the–art methodologies? Secondly, is a frequency-only representation enough to extract relevant features by the deep model?

We were able to improve on the results (Chapter 3, section Section 3.4) obtained using traditional ML and state-of-the-art strategies by combining spectrograms and a DL model. For a 2-class MI dataset (BCI competition III dataset IIIa) with the 5 layer dCNN we were able to see an average model accuracy of 91.81%, across the participants, as seen in Table 3.1. The results were taken as an average over 5 k-folds. This exceeded the performance of the ResNet50 network, which averaged classification accuracy of 87.85%. Table 3.2 shows the results of the same model and framework used for the full four classes in the BCI competition III dataset IIIa. We found the classification accuracy at an average of 90.81% across the participants, also performing better than state-of-the-art model.

Experiments demonstrated that, despite the model's high accuracy across the whole dataset, the framework were not robust across different datasets. This framework worked effectively over individuals in a dataset, however was not able to generalise over the whole dataset. Re-

sults are shown in Table 3.3 for the performance of the model on the overall BCI competition III dataset IIIa (Chapter 3, Section 3.4), for binary classification (left hand, right hand). The proposed framework with the 5-layer dCNN resulted in a mean of 74.95% across 5 k-folds. Though the classification accuracy is seen to be less than optimal, there is a improvement of 2.95% from the highest performing state-of-the-art model. This raised the question of whether we would see an improvement in results if we could retain the spatial, temporal, and spectral information contained within the data. The answer was favourable. As a result, the data pipeline was designed to augment the data utilised to generate topographical image projections. This drove the research for the spectrograms chapters, and the questions posed, also mentioned in the Introduction chapter, (Section 1.2). First, would providing a topographical representation of MI EEG signals to a DNN improve on results obtained from classical ML using state–of–the–art methodologies? Second, are we able to find a robust model which is able to generalise over all the subjects, and would not be required to train on an individual subject-to-subject basis? To begin, we utilised this framework to identify models that performed well on a two-class dataset and subsequently on an individual level dataset. Following that, experiments were conducted to assess whether the model was generalisable across the complete dataset in its entirety.

The study's findings demonstrated that the dCNN model was capable of generalising with a high degree of accuracy and robustness across the entire dataset of two-class datasets. Overall, we performed experiments on a total of 6 datasets. This included three 2-class MI datasets, two 3-class datasets and finally one 4-class dataset. As demonstrated in the results section of the previous chapter(Chapter 4, Section 4.5), when compared to current state-of-the-art ML techniques and model accuracy, the topographical maps - dCNN framework was extremely successful. The success of this DL framework was echoed when compared to other existing frameworks in the field.

The implementation consisted of a 9-layer dCNN network, and inputs of the topographical maps image data. The model was trained and optimised on the 52 participant GigaGB dataset (Section 4.5.1). This two-class MI dataset, performed with an accuracy of 94.12%

and outperformed the compared state of the art model as seen in table 4.2. We then went on to compare it with two further two-class MI datasets BCI Competition III dataset IVA (right hand, left foot MI), and BCI IV dataset I (right hand, left hand MI). The former, BCI III-dataset IVA showed performed poorly with an average accuracy across the dataset of 60.10% as shown in Table 4.4, this poor result was reflected also with the compared state-of-the-art model. However the proposed model performed better by an average of 5.36% over the 5-folds. We attributed this poor performance to this dataset being for right hand and left foot MI, where the original model was trained on a dataset for right hand and left hand MI data. This was reinforced with BCI IV dataset I which has right hand and left hand MI data. As seen in Section4.5.3, Table 4.6, classification accuracy across the 5-folds was 98.48%.

We tested the same model on three- and four-class datasets, respectively. Section 4.5.5, Table 4.10, shows the results of the BCI Competition III dataset V which consisted of left hand, right hand and word image MI, had an average accuracy of 99.62% classification accuracy over 5 k-folds using the same 9-layer dCNN network framework as described above. This performed with a much greater classification accuracy than state-of-the-art-model, which showed performance of 34.93% as seen in Section 4.5.5, Table 4.7. In contrast, BCI competition III dataset IVB which consisted of right hand, foot and relax states performed with a classification accuracy of 40.39%; however this performance is higher than that of ResNet50 which averaged at 37.17%. The conjecture, as reflected in the binary class MI dataset BCI III-dataset IVA, was that due to the original dCNN model being optimised on the binary (right hand, left hand dataset), the model is not able to discriminate much beyond these parameters. The four class dataset (results shown in Section 4.5.6, Table 4.12), BCI competition III dataset IIIa which consisted of the MI data of feet, left hand, right hand, tongue, showed poor performance of 41.67%. We attribute this poor performance to the added parameters beyond the right and left hand data that the model was optimised to.

There could be a multitude of reasons for the low classification accuracy. To begin, these datasets include much fewer participants and data points than their two-class equivalents. Also, the dCNN model may not be suitable for all classification levels. By increasing the

number of hidden layers, it is possible that the viability of the model would increase. Additionally, it is possible that we are limited in our view of the data within due to the image made by the pipeline $(28x28)$ and the image created by the pipeline. This would be a potential avenue for future research. Increased image resolution may enable us to extract more information from the data while preserving the dCNN architecture currently proposed. On the other hand, these solutions result in increased computing time and power usage.

In conclusion, this study demonstrates that for two-class MI datasets, the proposed 9-layer dCNN showed consistent high classification accuracy performance, with the capability to generalise over the entire dataset. The performed experiments suggest that it is pertinent to conclude that using topographical DL solutions are capable of learning to generalise over multiple users and can distinguish particularly well between left and right hand MI. The performance for both individual subjects as well as intra-subject classification accuracy decreased. The robustness of the model across entire datasets is also set to decrease given an increase in the number of classes. It should be noted, that this model performed consistently with a higher classification accuracy with comparison to state-of-the-art-models.

5.2 Challenges

Although research into BCI technology has been garnering attention for the last two decades, these technologies have primarily remained isolated in a research environment and have yet to make their way into clinical and home settings.

This section highlights the significant challenges that have prevented the general use of BCIs, which have been classified into five categories: challenges encountered during research and development of BCIs, the obstacles impeding commercialisation, flaws in testing approaches that have been identified in the literature, issues that have been encountered during BCI use that may prevent their widespread adoption, and the ethical considerations which should be taken into account.

5.2.1 Challenges in research and development

There are numerous signal processing issues involved in the research and development of BCI interfaces, particularly those that are based on MI EEG. These include determining the most effective strategies for feature extraction and selection, which is difficult due to the extremely non-linear, non-stationary, and artefact-prone nature of EEG data, as well as identifying the most effective techniques for feature extraction and selection. There are also issues in data fusion, namely in determining how the data from multiple EEG channels may be integrated in order to reduce data dimensionality while also potentially improving the classification results.

Further research is also required in order to determine the most appropriate categorisation approaches for the given features. Identifying the optimal approaches to utilise for patients with spinal cord injury or sickness should be a primary focus of research into features and classifiers, as the MI-EEG characteristics of such patients can differ from those of healthy persons. In addition, more effective training methodologies must be investigated through research. It is customary for a BCI to be used by a specific subject to require a significant number of training attempts from that particular subject, resulting in the calibration stage taking an excessively long time for a practical system.

As a result, investigations aimed at decreasing the calibration time are required. In an attempt to reduce the training time, the covariance matrices associated with CSP features collected from EEG trials to aid in the decoding of EEG signals have been used to aid in the decoding of EEG signals [179, 180, 181]. However, despite the fact that the geometry of the covariance matrix can be used to extract salient information from EEG data [182], current algorithms do not take advantage of it. It has been demonstrated that the geometric qualities of covariance matrices exist in symmetrical positive definite space, and Singh et al. [182] have devised a framework that takes advantage of SPD characteristics to reduce calibration time. On the IVa dataset, this framework outperformed other approaches that had previously been tried. Finally, one of the most difficult challenges in the study and development

of BCIs is the design of trustworthy systems with stable performance that can be utilised by a wide variety of users in various mental states and various locations.

5.2.2 Challenges in commercialisation

Challenges of BCIs is hindered by two significant barriers: the first is technical, which can be overcome through the development of more robust, efficient, and accurate signal processing infrastructures, and the second is the adaptation of laboratory-based technologies for use in the real world. Both of these issues are discussed in greater detail in the sections that follow.

Vansteensel et al. [183] distributed a questionnaire to over 3500 BCI researchers worldwide, 95% of whom worked on BCIs based on EEG or EMG technologies, to gain a clear picture of the barriers impeding the commercialisation of BCI interfaces. The researchers who replied to the poll judged BCIs, particularly those meant to replace or augment brain activity, to be practical and commercially viable. They predicted they would be commercially viable and feasible within the next five to ten years. According to the survey's findings, significant technological advancements in sensor technology, overall system performance, and user-friendliness were required for non-invasive BCIs. In contrast, for invasive BCIs, it was necessary to develop wholly implantable systems, improve system robustness and performance, and conduct clinical trials to ensure the systems' safety.

Additionally, the authors recommended that future research in BCI technology should focus on increasing bit rates [184], refining signal processing approaches, and experimenting with alternative categorisation methodologies. The authors propose that new approaches for designing overarching BCI systems, as well as the types of control systems used, be established at the macroeconomic level.

One of the most significant advancements in the commercialisation of BCIs has been the adaptation of laboratory-based interfaces for usage in the real world. Although BCIs have the potential to be applied to a variety of fields, including home automation [185, 186] ,

prosthetics [10, 11, 187, 188], rehabilitation [189, 190, 191], gaming [11, 192, 193, 194], transportation [195, 196] , education [197] , virtual reality [198], artistic computing [199, 200], and possibly even virtual assistants based on affective computing [199, 200], the leap to creating.

These considerations include, but are not limited to, the following: (i) technology selection, (ii) broad appeal, (iii) intuitivism, (iv) usability and reliability, and (v) cost [10, 201, 202, 203, 204]. This section will cover each of these elements in further depth. However, when designers make decisions about any of these variables, it is critical to consider the specific circumstance and setting in which the technology will be employed. For instance, the design requirements for an economical technology for controlling a television in a home environment may be different from those for a BCI system monitoring the attentiveness of a pilot in a plane. Similarly, designers must consider the users' health; technologies meant for rehabilitation or restoring lost CNS functionality frequently have additional or different needs than technologies intended for healthy users. Thus, the engineers must conduct an in-depth examination of the system's fundamental requirements throughout the early stages of system design, considering the environment, circumstance, and target audience.

The initial and most critical stage in building a human-computer interface, particularly one designed for use in factual circumstances, is selecting the appropriate technology. The technology selection process begins with examining all available hardware alternatives for the interface, including EEG, EMG, EOG, and other forms of eye-gaze tracking technology, as well as hybrid combinations of these technologies. At this point, it may be determined that a BCI is not the best solution for the intended commercial use and that another form of technology should be examined. When evaluating a BCI technology, a choice between an evoked and spontaneous system must be made, and the trade-offs between these two technologies must be considered.

The following topics are predicated on the use of an EEG-based device. For instance, MI-based technology may be well-suited for controlling a prosthetic limb or a small robotic arm, as the concept of body movement is natural in this case. However, for digital radio control,

an SSVEP-based system with a four-option menu comprising of station-up, station-down, volume-up, and volume-down possibilities may be more intuitive than an MI-based system with imagined movements associated with television controls.

This element of intuitivism should be considered during the first decision process when technology and BCI type are chosen. General appeal is another factor that underpins all design process areas; it encompasses anything that may influence the user's initial opinion of the system. The system's portability is critical: Bluetooth or Wi-Fi-enabled headsets enable users to walk around freely while still utilising the system, as opposed to a corded headset that restricts movement. The aesthetics of the headset and the graphical user interfaces employed would also influence the adoption of the technology.

Additionally, a choice can be made about the type of electrodes to be used: wet or dry. Dry electrodes are considered a more effective solution because they eliminate the inconvenience of inserting electroconductive gel between the scalp electrodes and leave no remnant of gel in the hair after use. However, there is controversy over whether dry electrodes provide the same signal quality as wet electrodes [205], with some research indicating that dry electrodes create noisier and more artefact-prone signals than wet electrodes [206]. Water-based electrodes have also been researched recently. It was observed in [207] that water- and dry-based electrodes performed comparably to gel-based electrodes in subjects with shorter hair, and it was suggested that with further refinement of the electrodes for subjects with varying hair lengths, water- and dry-based electrodes might be used instead of gel electrodes.

BCI systems must be highly usable and reliable in order to be successful in the market. Usability includes both design ergonomics and simplicity of use and the time required to train a new user on the system, which should be low. The ideal technology would be that ordinary people could pick up and learn to use intuitively or through brief instruction, similar to how they would when purchasing a new phone. Additionally, the systems must be user-friendly and incorporate protections to prevent unsafe use. Additionally, systems must be dependable, with users believing that the technology is trustworthy and produces consistent

outcomes past the initial learning curve of new technology. Additionally, the system must be reliable when utilised in the multimodal contexts in which it is intended to be used, such as a noisy family home, a crowded design studio, or the constantly changing environment of an emergency operating theatre.

These multimodal environments may affect expected brain responses compared to controlled laboratory recordings, necessitating extensive development and testing of such devices. The user's condition during use may also need to be considered, like heart rate and cortisol levels—a stress hormone—have been shown to interact with the EEG signal response [208]. Seo and Lee discovered a substantial positive connection between higher beta band power—a characteristic of MI EEG data—and cortisol level [209]. Finally, the cost is a considerable impediment. The typical budget of the projected end users and any prospective economies of scale should be considered early on since this will dictate the sorts of recording equipment and sensors that can be utilised, as well as the software and any compliance requirements specific to the target audience.

While the objective should always be to give the optimal trade-off between cost and performance, the price ranges within which students, families, private healthcare firms, the military, start-ups, and substantial technology corporations may operate varies just as widely as their needs. Researchers serious about developing new technologies for a specific audience would conduct market research before initiating the system's design process.

The commercialisation of BCIs would also necessitate the development of industry standards, particularly for benchmarking databases, EEG recording equipment, and software applications [13]. International roadmap initiatives like BNCI Horizon 2020 [10] have sought to strengthen collaboration between industry and research in order to tackle such difficulties.

5.2.3 Shortcomings in testing approaches

Extensive testing on large populations is required to implement these recommended improvements, but the testing process itself is often flawed. The literature contains a range of performance measures for evaluating BCIs, and the absence of a standardised technique or a single metric for quantifying a system's overall performance means that comparisons between systems are limited [210]. Furthermore, reporting on fundamental statistics such as accuracy may mask essential considerations in the BCI system, such as the trade-off between accuracy and speed [210, 211].

The flaws in widely used testing approaches are not restricted to the performance metrics employed but also to the data used for testing. Numerous BCIs intended to replace or restore CNS functionality are tested on healthy volunteers in a laboratory setting. However, if the typical end-users of these systems are people with functional disabilities, this can result in unrealistically optimistic findings.

A comprehensive database of MI EEG data from such patients and healthy controls performing the same tasks would provide an abundance of data for research, permitting the resilience of technologies to be tested and new solutions to be identified if they fail users with particular disabilities. Even data from healthy patients are frequently collected in carefully controlled laboratory situations. While this data is critical for the first evaluation of signal processing techniques, creating more robust and commercially viable systems would require stress testing. As previously stated, heart rate and cortisol levels can alter the quality of EEG signals in contexts outside the lab, both indoors and outdoors, with diverse sensory input from noises, movements, and scents, and with participants seated in a variety of positions.

A database of subjects performing the same set of MI tasks in all of these different scenarios, as well as in a controlled laboratory setting, would be an excellent starting point for assessing how the effectiveness of MI data processing techniques varies across scenarios and how

these techniques can be made more robust.

5.2.4 Issues preventing wide-spread adoption

A significant hurdle in the face of widespread adoption of BCI technologies is BCI illiteracy. This arises when a user cannot control a BCI due to their inability to produce the needed high-quality brain signals [13]. The quality of the EEG signal and the user's overall mastery of the BCI can be improved by employing an interactive, co-learning approach that gives feedback, such as auditory or visual input, while the user interacts with the system [212].

Long-term use of BCIs requires repeated usage of specific neural pathways, and additional research is needed to determine any potential health consequences or changes in brain function. For example, long-term usage of external actuators via BCIs has been shown to result in a reorganisation of the brain's map of the body, with the actuators being interpreted as an extension of the subject's body [13, 213].

5.2.5 Ethical considerations

To guide the future development of BCI technology, ethical norms must also be created. Such ethics would define accountability for accidents that occur when using controlled apparatus and address the acceptable use of bio-signal data in the context of privacy. The BCI Society seeks to address some of these demands through the publication of ethical principles and guidelines [11, 10].

5.3 Future Work

The next steps in improving the proposed processing pipeline is to offer more attention to channel selection. A 2018 paper [214] looked at studying EEG signals in patients with

brain death. The channel reduction applied in [134], was also applied in [214], in which the $C3$, $C4$ and Cz electrodes, shown in Fig. 2.6, were used in data analysis. Here there was also the use of STFT, in particular looking for the mu and beta band of frequencies. Their research involved an analysis of the signals from the motor cortex using a reduced number of channels. Wang et al. [137] have also conducted studies on the motor cortex using STFT. Their analysis is only focused on 10 centrally located channels focusing on the motor cortex. Additionally, Wang et al. [137] have also trained their models separately for individual users, with a wide range of variability in accuracy. Reducing the number of channels has the potential to improve the performance of the NN, as additional channels introduce uncorrelated data from regions of the cortex not directly responsible for the task under scrutiny.

The main goal of BCI is to empower people with disabilities. Therefore mirroring this it is very important for response of the BCI to be as accurate as possible such that the system feedback is perceived as natural. Working towards this would require further data collections, including an online data collection with feedback. It would additionally be appropriate to train an additional class which would be a state of relaxation. This pollutes the search space with noise and reduce the useful information available to the network. This could also have a practical application for data collection. The complexity of setting up the head–set along with difficulties in maintaining the electrodes from drying by continuously applying gel can affect the quality of the data. Additionally, the proximity of the electrodes to the facial muscles could also introduce greater variability in the signal, as well as drawing away focus from the actual regions of interest, namely the motor cortex, which occupies a more central region of the scalp.

Existing BCI systems rely primarily on volunteers with typical cognitive abilities, while some applications require unique populations such as the elderly and those with neurological disorders. Numerous experiments will be required to determine the individual distinctions between a normally functioning person and a disabled or elderly person in order to apply BCI systems as a service.

DL is a type of "Black Box" ML, and thus it is difficult to understand reasoning behind the predictions which are made by the model in question. In recent times, the importance of understanding this reasoning has become a growing area of research, and is known as "Interpretable Machine Learning" [215, 216]. This area will be critical in gaining a better understanding of how deep networks work.

The research of MI BCIs in assistive technologies has a great deal to gain from research into DNNs and their internal mechanisms. A thorough understanding of these networks could lead to the development of a motor-imagery BCI that is fully robust and reliable and is capable of offering accurate assistance to people with disabilities. The functionality and impact of assistive technologies can be greatly enhanced by further research in this area, which could completely alter how people with disabilities interact with their surroundings. Thus, the study of DNNs and their applications in BCIs is a crucial area of research that holds great promise for the future.

Bibliography

[1] J. Bickenbach, "The world report on disability," *Disability & Society*, vol. 26, no. 5, pp. 655–658, 2011.

[2] "Department of work and pensions (gov.uk) - family resources survey: financial year 2019 to 2020," https://www.gov.uk/government/statistics/family-resources-survey-financial-year-2019-to-2020/family-resources-survey-financial-year-2019-to-2020, 2020, (Accessed on 06/11/2021).

[3] "The statistics of disabled people living in the united kingdom," https://www.dialuk.info/statistics-of-disabled-people-united-kingdom, (Accessed on 06/11/2021).

[4] M. Soegaard and R. F. Dam, *Brief intro. in the encyclopedia of human-computer interaction*. Interaction Design Foundation, 2013.

[5] B. Allison, B. Graimann, and G. Pfurtscheller, *Brain-computer Interfaces: Revolutionizing Human-computer Interaction*. Springer, 2010.

[6] M. Sharanreddy and P. K. Kulkarni, "Automated EEG signal analysis for identification of epilepsy seizures and brain tumour," *Journal of Medical Engineering & Technology*, vol. 37, no. 8, pp. 511–519, Oct. 2013. [Online]. Available: https://doi.org/10.3109/03091902.2013.837530

[7] J. Kevric and A. Subasi, "Comparison of signal decomposition methods in classification of eeg signals for motor-imagery bci system," *Biomedical Signal Processing and Control*, vol. 31, pp. 398–406, 2017.

[8] A. J. Casson, "Wearable EEG and beyond," *Biomedical Engineering Letters*, vol. 9, no. 1, pp. 53–71, Jan. 2019. [Online]. Available: https://doi.org/10.1007/s13534-018-00093-6

[9] V. P. Oikonomou, K. Georgiadis, G. Liaros, S. Nikolopoulos, and I. Kompatsiaris, "A comparison study on eeg signal processing techniques using motor imagery eeg data," in *2017 IEEE 30th international symposium on computer-based medical systems (CBMS)*. IEEE, 2017, pp. 781–786.

[10] G. Kristo, J. Höhne, R. Ortner, B. Reuderink, and N. Ramsey, "Contribution to roadmap - bnci horizon 2020," 2015.

[11] A. Nijholt, "The future of brain-computer interfacing," in *2016 5th International Conference on Informatics, Electronics and Vision (ICIEV)*. IEEE, 2016, pp. 156–161.

[12] S. Sakhavi, C. Guan, and S. Yan, "Learning temporal information for brain-computer interface using convolutional neural networks," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 11, pp. 5619–5629, 2018.

[13] N. Padfield, J. Zabalza, H. Zhao, V. Masero, and J. Ren, "Eeg-based brain-computer interfaces using motor-imagery: Techniques and challenges," *Sensors*, vol. 19, no. 6, p. 1423, 2019.

[14] A. Bohr and K. Memarzadeh, "The rise of artificial intelligence in healthcare applications," in *Artificial Intelligence in Healthcare*. Elsevier, 2020, pp. 25–60. [Online]. Available: https://doi.org/10.1016/b978-0-12-818438-7.00002-2

[15] G. Xu, X. Shen, S. Chen, Y. Zong, C. Zhang, H. Yue, M. Liu, F. Chen, and W. Che, "A deep transfer convolutional neural network framework for eeg signal classification," *IEEE Access*, vol. 7, pp. 112 767–112 776, 2019.

[16] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Medical image analysis*, vol. 42, pp. 60–88, 2017.

[17] I. Majidov and T. Whangbo, "Efficient classification of motor imagery electroencephalography signals using deep learning methods," *Sensors*, vol. 19, no. 7, p. 1736, 2019.

[18] S. H. Khan, M. Hayat, and F. Porikli, "Regularization of deep neural networks with spectral dropout," *Neural Networks*, vol. 110, pp. 82–90, 2019.

[19] E. Nurvitadhi, G. Venkatesh, J. Sim, D. Marr, R. Huang, J. Ong Gee Hock, Y. T. Liew, K. Srivatsan, D. Moss, S. Subhaschandra *et al.*, "Can fpgas beat gpus in accelerating next-generation deep neural networks?" in *Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*, 2017, pp. 5–14.

[20] S. Yang, S. Lopez, M. Golmohammadi, I. Obeid, and J. Picone, "Semi-automated annotation of signal events in clinical eeg data," in *2016 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*. IEEE, 2016, pp. 1–5.

[21] A. Puce and M. S. Hämäläinen, "A review of issues related to data acquisition and analysis in eeg/meg studies," *Brain sciences*, vol. 7, no. 6, p. 58, 2017.

[22] M. Rashid, N. Sulaiman, A. PP Abdul Majeed, R. M. Musa, B. S. Bari, S. Khatun *et al.*, "Current status, challenges, and possible solutions of eeg-based brain-computer interface: a comprehensive review," *Frontiers in neurorobotics*, vol. 14, p. 25, 2020.

[23] C. M. Signorelli, "Can computers become conscious and overcome humans?" *Frontiers in Robotics and AI*, vol. 5, p. 121, 2018.

[24] T. H. Davenport, "From analytics to artificial intelligence," *Journal of Business Analytics*, vol. 1, no. 2, pp. 73–80, 2018.

[25] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: A brief review," *Computational intelligence and neuroscience*, vol. 2018, 2018.

[26] D. F. Wulsin, J. R. Gupta, R. Mani, J. A. Blanco, and B. Litt, "Modeling electroencephalography waveforms with semi-supervised deep belief nets: Fast classification and anomaly measurement," *Journal of Neural Engineering*, vol. 8, no. 3, p. 036015, jun 2011. [Online]. Available: /pmc/articles/PMC3193936//pmc/articles/PMC3193936/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC3193936/

[27] D. L. Schomer and F. H. L. da Silva, Eds., *Niedermeyers Electroencephalography*. Oxford University Press, Nov. 2017. [Online]. Available: https://doi.org/10.1093/med/9780190228484.001.0001

[28] G. Pfurtscheller and C. Neuper, "Motor imagery and direct brain-computer communication," *Proceedings of the IEEE*, vol. 89, no. 7, pp. 1123–1134, Jul. 2001. [Online]. Available: https://doi.org/10.1109/5.939829

[29] 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, 2012. [Online]. Available: https://doi.org/10.3389/fnins.2012.00039

[30] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, and K. robert Muller, "Optimizing spatial filters for robust EEG single-trial analysis," *IEEE Signal Processing Magazine*, vol. 25, no. 1, pp. 41–56, 2008. [Online]. Available: https://doi.org/10.1109/msp.2008.4408441

[31] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, "EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces," *Journal of Neural Engineering*, vol. 15, no. 5, p. 056013, Jul. 2018. [Online]. Available: https://doi.org/10.1088/1741-2552/aace8c

[32] Y.-L. Tseng, C.-S. Kuo, and M. Liou, "Event-related spectral perturbations and phase synchronization of EEG during natural auditory memory retrieval task," in *2014 IEEE International Symposium on Bioelectronics and Bioinformatics (IEEE ISBB 2014)*. IEEE, Apr. 2014. [Online]. Available: https://doi.org/10.1109/isbb.2014.6820929

[33] A. H. Mooij, B. Frauscher, M. Amiri, W. M. Otte, and J. Gotman, "Differentiating epileptic from non-epileptic high frequency intracerebral EEG signals with measures of wavelet entropy,"

Clinical Neurophysiology, vol. 127, no. 12, pp. 3529–3536, Dec. 2016. [Online]. Available: https://doi.org/10.1016/j.clinph.2016.09.011

[34] D. A. Bridwell, J. F. Cavanagh, A. G. E. Collins, M. D. Nunez, R. Srinivasan, S. Stober, and V. D. Calhoun, "Moving beyond ERP components: A selective review of approaches to integrate EEG and behavior," *Frontiers in Human Neuroscience*, vol. 12, Mar. 2018. [Online]. Available: https://doi.org/10.3389/fnhum.2018.00106

[35] B. J. Roach and D. H. Mathalon, "Event-related EEG time-frequency analysis: An overview of measures and an analysis of early gamma band phase locking in schizophrenia," *Schizophrenia Bulletin*, vol. 34, no. 5, pp. 907–926, Jul. 2008. [Online]. Available: https://doi.org/10.1093/schbul/sbn093

[36] O. Dressler, G. Schneider, G. Stockmanns, and E. Kochs, "Awareness and the EEG power spectrum: analysis of frequencies," *British Journal of Anaesthesia*, vol. 93, no. 6, pp. 806–809, Dec. 2004. [Online]. Available: https://doi.org/10.1093/bja/aeh270

[37] Q. Liu, J. Cai, S.-Z. Fan, M. F. Abbod, J.-S. Shieh, Y. Kung, and L. Lin, "Spectrum analysis of eeg signals using cnn to model patient's consciousness level based on anesthesiologists' experience," *IEEE Access*, vol. 7, pp. 53 731–53 742, 2019.

[38] R. G. Real, B. Kotchoubey, and A. Kübler, "Studentized continuous wavelet transform (t-cwt) in the analysis of individual erps: real and simulated eeg data," *Frontiers in neuroscience*, vol. 8, p. 279, 2014.

[39] Z. Dokur and T. Olmez, "Classification of motor imagery eeg signals by using a divergence based convolutional neural network," *arXiv preprint arXiv:2103.10977*, 2021.

[40] C. D. V. Gonzalez, J. H. S. Azuela, E. R. Espino, and V. H. P. Ponce, "Classification of motor imagery eeg signals with csp filtering through neural networks models," in *Mexican International Conference on Artificial Intelligence*. Springer, 2018, pp. 123–135.

[41] X. Gui, C. Chuansheng, L. Zhong-Lin, and D. Qi, "Brain imaging techniques and their applications in decision-making research," *Xin li xue bao. Acta psychologica Sinica*, vol. 42, no. 1, p. 120, 2010.

[42] K.-W. Ha and J.-W. Jeong, "Motor imagery eeg classification using capsule networks," *Sensors*, vol. 19, no. 13, p. 2854, 2019.

[43] G. E. Chatrian, M. C. Petersen, and J. A. Lazarte, "The blocking of the rolandic wicket rhythm and some central changes related to movement," *Electroencephalography and clinical neurophysiology*, vol. 11, no. 3, pp. 497–510, 1959.

[44] 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.

[45] B. Lei, X. Liu, S. Liang, W. Hang, Q. Wang, K.-S. Choi, and J. Qin, "Walking imagery evaluation in brain computer interfaces via a multi-view multi-level deep polynomial network," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 27, no. 3, pp. 497–506, 2019.

[46] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai *et al.*, "Recent advances in convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 354–377, 2018.

[47] M. Alansari, M. Kamel, B. Hakim, and Y. Kadah, "Study of wavelet-based performance enhancement for motor imagery brain-computer interface," in *2018 6th International Conference on Brain-Computer Interface (BCI)*. IEEE, 2018, pp. 1–4.

[48] M. Aljalal and R. Djemal, "A comparative study of wavelet and csp features classified using lda, svm and ann in eeg based motor imagery," in *2017 9th IEEE-GCC Conference and Exhibition (GCCCE)*. IEEE, 2017, pp. 1–9.

[49] M. Behri, A. Subasi, and S. M. Qaisar, "Comparison of machine learning methods for two class motor imagery tasks using eeg in brain-computer interface," in *2018 Advances in Science and Engineering Technology International Conferences (ASET)*. IEEE, 2018, pp. 1–5.

[50] M. Mirnaziri, M. Rahimi, S. Alavikakhaki, and R. Ebrahimpour, "Using combination of μ, β and γ bands in classification of eeg signals," *Basic and clinical neuroscience*, vol. 4, no. 1, p. 76, 2013.

[51] V. Mishuhina and X. Jiang, "Feature weighting and regularization of common spatial patterns in eeg-based motor imagery bci," *IEEE Signal Processing Letters*, vol. 25, no. 6, pp. 783–787, 2018.

[52] B. Li, B. Yang, C. Guan, and C. Hu, "Three-class motor imagery classification based on fbcsp combined with voting mechanism," in *2019 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*. IEEE, 2019, pp. 1–4.

[53] M. K. I. Molla, A. Al Shiam, M. R. Islam, and T. Tanaka, "Discriminative feature selection-based motor imagery classification using eeg signal," *IEEE Access*, vol. 8, pp. 98 255–98 265, 2020.

[54] V. Silva, R. Barbosa, P. Vieira, and C. Lima, "Ensemble learning based classification for bci applications, ieee 5th portuguese meeting on bioengineering," 2017.

[55] Y. Wang, X. Gao, B. Hong, C. Jia, and S. Gao, "Brain-computer interfaces based on visual evoked potentials," *IEEE Engineering in medicine and biology magazine*, vol. 27, no. 5, pp. 64–71, 2008.

[56] J. Wang, Z. Feng, and N. Lu, "Feature extraction by common spatial pattern in frequency domain for motor imagery tasks classification," in *2017 29th Chinese Control and Decision Conference (CCDC)*. IEEE, 2017, pp. 5883–5888.

[57] Y. Zhang, J. Liu, J. Liu, J. Sheng, and J. Lv, "Eeg recognition of motor imagery based on svm ensemble," in *2018 5th International Conference on Systems and Informatics (ICSAI)*. IEEE, 2018, pp. 866–870.

[58] X. Yong and C. Menon, "Eeg classification of different imaginary movements within the same limb," *PloS one*, vol. 10, no. 4, p. e0121896, 2015.

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

[60] J. E. Hall, *Guyton and Hall textbook of medical physiology e-Book*. Elsevier Health Sciences, 2015.

[61] P. Prashant, A. Joshi, and V. Gandhi, "Brain computer interface: A review," in *Engineering (NUiCONE), 2015 5th Nirma University International Conference on*. IEEE, 2015, pp. 1–6.

[62] A. Al-Saegh, S. A. Dawwd, and J. M. Abdul-Jabbar, "Deep learning for motor imagery EEG-based classification: A review," p. 102172, jan 2021.

[63] L. F. Nicolas-Alonso and J. Gomez-Gil, "Brain computer interfaces, a review," *sensors*, vol. 12, no. 2, pp. 1211–1279, 2012.

[64] P. Stawicki, F. Gembler, A. Rezeika, and I. Volosyak, "A novel hybrid mental spelling application based on eye tracking and ssvep-based bci," *Brain sciences*, vol. 7, no. 4, p. 35, 2017.

[65] B. He, H. Yuan, J. Meng, and S. Gao, "Brain–computer interfaces," in *Neural engineering*. Springer, 2020, pp. 131–183.

[66] A. Vourvopoulos, S. Bermudez i Badia, and F. Liarokapis, "Eeg correlates of video game experience and user profile in motor-imagery-based brain–computer interaction," *The Visual Computer*, vol. 33, no. 4, pp. 533–546, 2017.

[67] 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.

[68] J. P. Donoghue, "Connecting cortex to machines: recent advances in brain interfaces," *Nature neuroscience*, vol. 5, no. 11, pp. 1085–1088, 2002.

[69] Y. Li, K. K. Ang, and C. Guan, "Digital signal processing and machine learning," in *Brain-Computer Interfaces*. Springer Berlin Heidelberg, 2009, pp. 305–330. [Online]. Available: https://doi.org/10.1007/978-3-642-02091-9_17

[70] A. Bashashati, M. Fatourechi, R. K. Ward, and G. E. Birch, "A survey of signal processing algorithms in brain–computer interfaces based on electrical brain signals," *Journal of Neural engineering*, vol. 4, no. 2, p. R32, 2007.

[71] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi, "A review of classification algorithms for eeg-based brain–computer interfaces," *Journal of neural engineering*, vol. 4, no. 2, p. R1, 2007.

[72] S. G. Mason and G. E. Birch, "A general framework for brain-computer interface design," *IEEE transactions on neural systems and rehabilitation engineering*, vol. 11, no. 1, pp. 70–85, 2003.

[73] 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.

[74] H. K. Lee and Y.-S. Choi, "Application of continuous wavelet transform and convolutional neural network in decoding motor imagery brain-computer interface," *Entropy*, vol. 21, no. 12, p. 1199, 2019.

[75] 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.

[76] "Human physiology."

[77] M. Teplan *et al.*, "Fundamentals of eeg measurement," *Measurement science review*, vol. 2, no. 2, pp. 1–11, 2002.

[78] J. D. Kropotov and A. Muller, "What can event related potentials contribute to neuropsychology," *Acta Neuropsychologica*, vol. 7, no. 3, pp. 169–81, 2009.

[79] K. J. Miller, D. Hermes, and N. P. Staff, "The current state of electrocorticography-based brain–computer interfaces," *Neurosurgical focus*, vol. 49, no. 1, p. E2, 2020.

[80] F. L. da Silva, S. Gonçalves, and J. D. Munck, "Electroencephalography (EEG)," in *Encyclopedia of Neuroscience*. Elsevier, 2009, pp. 849–855. [Online]. Available: https://doi.org/10.1016/b978-008045046-9.00304-1

[81] M. H. Díaz, F. M. Córdova, L. Cañete, F. Palominos, F. Cifuentes, C. Sánchez, and M. Herrera, "Order and chaos in the brain: fractal time series analysis of the eeg activity during a cognitive problem solving task," *Procedia Computer Science*, vol. 55, pp. 1410–1419, 2015.

[82] A. Schlögl, F. Lee, H. Bischof, and G. Pfurtscheller, "Characterization of four-class motor imagery eeg data for the bci-competition 2005," *Journal of neural engineering*, vol. 2, no. 4, p. Lv14, 2005.

[83] M. Sazgar and M. G. Young, "Overview of eeg, electrode placement, and montages," in *Absolute Epilepsy and EEG Rotation Review*. Springer, 2019, pp. 117–125.

[84] W. Qiao and X. Bi, "Deep spatial-temporal neural network for classification of eeg-based motor imagery," in *Proceedings of the 2019 International Conference on Artificial Intelligence and Computer Science*, 2019, pp. 265–272.

[85] D. Zhang, L. Yao, K. Chen, S. Wang, X. Chang, and Y. Liu, "Making sense of spatio-temporal preserving representations for eeg-based human intention recognition," *IEEE transactions on cybernetics*, vol. 50, no. 7, pp. 3033–3044, 2019.

[86] K. Elf, E. Ronne-Engström, R. Semnic, E. Rostami-Berglund, J. Sundblom, and M. Zetterling, "Continuous eeg monitoring after brain tumor surgery," *Acta neurochirurgica*, vol. 161, no. 9, pp. 1835–1843, 2019.

[87] R. Pandya, S. Nadiadwala, R. Shah, and M. Shah, "Buildout of methodology for meticulous diagnosis of k-complex in eeg for aiding the detection of alzheimer's by artificial intelligence," *Augmented Human Research*, vol. 5, no. 1, pp. 1–8, 2020.

[88] E. Bresch, U. Grossekathofer, and G. Garcia-Molina, "Recurrent deep neural networks for real-time sleep stage classification from single channel eeg," *Frontiers in computational neuroscience*, vol. 12, p. 85, 2018.

[89] J. H. JW Britton, LC Frey, *Electroencephalography (EEG) : an introductory text and atlas of normal and abnormal findings in adults, children, and infants*. Chicago, IL: American Epilepsy Society, 2016.

[90] A. S. Al-Fahoum and A. A. Al-Fraihat, "Methods of eeg signal features extraction using linear analysis in frequency and time-frequency domains," *ISRN neuroscience*, vol. 2014, 2014.

[91] M. Z. Parvez and M. Paul, "Eeg signal classification using frequency band analysis towards epileptic seizure prediction," in *16th Int'l Conf. Computer and Information Technology*. IEEE, 2014, pp. 126–130.

[92] G. Dumas, J. Nadel, R. Soussignan, J. Martinerie, and L. Garnero, "Inter-brain synchronization during social interaction," *PloS one*, vol. 5, no. 8, p. e12166, 2010.

[93] L. E. Ismail and W. Karwowski, "Applications of eeg indices for the quantification of human cognitive performance: A systematic review and bibliometric analysis," *PloS one*, vol. 15, no. 12, p. e0242857, 2020.

[94] M. Shahriari, D. Feiz, A. Zarei, and E. Kashi, "The meta-analysis of neuro-marketing studies: past, present and future," *Neuroethics*, vol. 13, no. 3, pp. 261–273, 2020.

[95] S. Vaid, P. Singh, and C. Kaur, "Eeg signal analysis for bci interface: A review," in *2015 fifth international conference on advanced computing & communication technologies*. IEEE, 2015, pp. 143–147.

[96] "Emotiv - brain data measuring hardware and software solutions," https://www.emotiv.com/, (Accessed on 06/15/2021).

[97] "Neurosky eeg headsets," http://neurosky.com/, (Accessed on 06/15/2021).

[98] E. H. T. Shad, M. Molinas, and T. Ytterdal, "Impedance and noise of passive and active dry eeg electrodes: A review," *IEEE Sensors Journal*, vol. 20, no. 24, pp. 14 565–14 577, 2020.

[99] M. A. Lopez-Gordo, D. Sanchez-Morillo, and F. P. Valle, "Dry eeg electrodes," *Sensors*, vol. 14, no. 7, pp. 12 847–12 870, 2014.

[100] J. Xu, S. Mitra, C. Van Hoof, R. F. Yazicioglu, and K. A. Makinwa, "Active electrodes for wearable eeg acquisition: Review and electronics design methodology," *IEEE reviews in biomedical engineering*, vol. 10, pp. 187–198, 2017.

[101] M. Rabovsky, W. Sommer, and R. A. Rahman, "Depth of conceptual knowledge modulates visual processes during word reading," *Journal of Cognitive Neuroscience*, vol. 24, no. 4, pp. 990–1005, apr 2012. [Online]. Available: https://doi.org/10.1162/jocn_a_00117

[102] M. Mohri, *Foundations of machine learning*. Cambridge, MA: MIT Press, 2012.

[103] H. Cecotti and A. Graser, "Convolutional neural networks for p300 detection with application to brain-computer interfaces," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 3, pp. 433–445, 2010.

[104] K. Maurer and T. Dierks, *Atlas of brain mapping: topographic mapping of EEG and evoked potentials*. Springer Science & Business Media, 2012.

[105] M. K. Islam, A. Rastegarnia, and Z. Yang, "Les méthodes de détection et de rejet d'artefact de l'EEG de scalp : revue de littérature," pp. 287–305, nov 2016.

[106] A. E. Selim, M. A. Wahed, and Y. M. Kadah, "Machine learning methodologies in p300 speller brain-computer interface systems," in *2009 National Radio Science Conference*. IEEE, 2009, pp. 1–9.

[107] G. Costantini, M. Todisco, D. Casali, M. Carota, G. Saggio, L. Bianchi, M. Abbafati, and L. Quitadamo, "Svm classification of eeg signals for brain computer interface," in *Neural Nets WIRN09*. IOS Press, 2009, pp. 229–233.

[108] F. Wang, C. Zhou, X. Hao, S. Wang, and G. Yang, "Bci control system for humanoid robot based on motor imaginary," in *2013 25th Chinese Control and Decision Conference (CCDC)*. IEEE, 2013, pp. 5140–5143.

[109] R. Singla and B. Haseena, "Comparison of ssvep signal classification techniques using svm and ann models for bci applications," *International Journal of Information and Electronics Engineering*, vol. 4, no. 1, p. 6, 2014.

[110] L. Sun, B. Jin, H. Yang, J. Tong, C. Liu, and H. Xiong, "Unsupervised eeg feature extraction based on echo state network," *Information Sciences*, vol. 475, pp. 1–17, 2019.

[111] C. J. Kelly, A. Karthikesalingam, M. Suleyman, G. Corrado, and D. King, "Key challenges for delivering clinical impact with artificial intelligence," *BMC medicine*, vol. 17, no. 1, pp. 1–9, 2019.

[112] R. Miotto, F. Wang, S. Wang, X. Jiang, and J. T. Dudley, "Deep learning for healthcare: review, opportunities and challenges," *Briefings in bioinformatics*, vol. 19, no. 6, pp. 1236–1246, 2018.

[113] J. F. Gao, Y. Yang, P. Lin, P. Wang, and C. X. Zheng, "Automatic removal of eye-movement and blink artifacts from eeg signals," *Brain topography*, vol. 23, no. 1, pp. 105–114, 2010.

[114] S. Cheng, J. Wang, L. Zhang, and Q. Wei, "Motion imagery-bci based on eeg and eye movement data fusion," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 12, pp. 2783–2793, 2020.

[115] S.-M. Zhou, J. Q. Gan, and F. Sepulveda, "Classifying mental tasks based on features of higher-order statistics from eeg signals in brain–computer interface," *Information Sciences*, vol. 178, no. 6, pp. 1629–1640, 2008.

[116] A. Subasi and M. I. Gursoy, "Eeg signal classification using pca, ica, lda and support vector machines," *Expert systems with applications*, vol. 37, no. 12, pp. 8659–8666, 2010.

[117] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, pp. 85–117, 2015.

[118] I. Ullah, M. Hussain, E. ul Haq Qazi, and H. Aboalsamh, "An automated system for epilepsy detection using EEG brain signals based on deep learning approach," *Expert Systems with Applications*, vol. 107, pp. 61–71, Oct. 2018. [Online]. Available: https://doi.org/10.1016/j.eswa.2018.04.021

[119] M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, M. Hasan, B. C. V. Essen, A. A. S. Awwal, and V. K. Asari, "A state-of-the-art survey on deep learning theory and architectures," *Electronics*, vol. 8, no. 3, p. 292, Mar. 2019. [Online]. Available: https://doi.org/10.3390/electronics8030292

[120] G. Xu, X. Shen, S. Chen, Y. Zong, C. Zhang, H. Yue, M. Liu, F. Chen, and W. Che, "A deep transfer convolutional neural network framework for EEG signal classification," *IEEE Access*, vol. 7, pp. 112 767–112 776, 2019. [Online]. Available: https://doi.org/10.1109/access.2019.2930958

[121] M. Dai, D. Zheng, R. Na, S. Wang, and S. Zhang, "EEG Classification of Motor Imagery Using a Novel Deep Learning Framework," *Sensors*, vol. 19, no. 3, p. 551, jan 2019. [Online]. Available: http://www.mdpi.com/1424-8220/19/3/551

[122] J. J. Escobar, J. Ortega, M. Damas, R. S. Kızıltepe, and J. Q. Gan, "Energy-time analysis of convolutional neural networks distributed on heterogeneous clusters for EEG classification," in *Advances in Computational Intelligence*. Springer International Publishing, 2019, pp. 895–907. [Online]. Available: https://doi.org/10.1007/978-3-030-20518-8_74

[123] J. Thomas, T. Maszczyk, N. Sinha, T. Kluge, and J. Dauwels, "Deep learning-based classification for brain-computer interfaces," in *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, Oct. 2017. [Online]. Available: https://doi.org/10.1109/smc.2017.8122608

[124] 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, Oct. 2019. [Online]. Available: https://doi.org/10.1088/1741-2552/ab3471

[125] S. U. Amin, M. Alsulaiman, G. Muhammad, M. S. Hossain, and M. Guizani, "Deep learning for eeg motor imagery-based cognitive healthcare," in *Connected Health in Smart Cities*. Springer, 2020, pp. 233–254.

[126] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, May 2017. [Online]. Available: https://doi.org/10.1145/3065386

[127] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks Networks for Large-Scale Image Recognition," in *International Conference on Learning Representations*, 2015. [Online]. Available: http://www.robots.ox.ac.uk/

[128] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[129] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[130] Z. Gao, X. Wang, Y. Yang, C. Mu, Q. Cai, W. Dang, and S. Zuo, "Eeg-based spatio–temporal convolutional neural network for driver fatigue evaluation," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 9, pp. 2755–2763, 2019.

[131] K. Yue and D. Wang, "Eeg-based 3d visual fatigue evaluation using cnn," *Electronics*, vol. 8, no. 11, p. 1208, 2019.

[132] H. Dose, J. S. Møller, S. Puthusserypady, and H. K. Iversen, "A deep learning mi-eeg classification model for bcis," in *2018 26th European Signal Processing Conference (EUSIPCO)*. IEEE, 2018, pp. 1676–1679.

[133] Y.-T. Wu, T. H. Huang, C. Y. Lin, S. J. Tsai, and P.-S. Wang, "Classification of EEG motor imagery using support vector machine and convolutional neural network," in *2018 International Automatic Control Conference (CACS)*. IEEE, nov 2018. [Online]. Available: https://doi.org/10.1109/cacs.2018.8606765

[134] Z. Wang, L. Cao, Z. Zhang, X. Gong, Y. Sun, and H. Wang, "Short time fourier transformation and deep neural networks for motor imagery brain computer interface recognition," *Concurrency and Computation: Practice and Experience*, vol. 30, no. 23, p. e4413, jan 2018. [Online]. Available: https://doi.org/10.1002/cpe.4413

[135] A. Supratak, H. Dong, C. Wu, and Y. Guo, "Deepsleepnet: A model for automatic sleep stage scoring based on raw single-channel eeg," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 11, pp. 1998–2008, 2017.

[136] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[137] H. Wang and W. Mo, "Motor imaginary eeg signals classification based on deep learning," in *International Conference on Bio-Inspired Computing: Theories and Applications*. Springer, 2018, pp. 142–150.

[138] Y. Hu, Y. Wong, W. Wei, Y. Du, M. Kankanhalli, and W. Geng, "A novel attention-based hybrid CNN-RNN architecture for sEMG-based gesture recognition," *PLOS ONE*, vol. 13, no. 10, p. e0206049, oct 2018. [Online]. Available: https://doi.org/10.1371/journal.pone.0206049

[139] S. A. Ludwig and J. Kong, "Investigation of different classifiers and channel configurations of a mobile p300-based brain–computer interface," *Medical & biological engineering & computing*, vol. 55, no. 12, pp. 2143–2154, 2017.

[140] X. Tang, W. Li, X. Li, W. Ma, and X. Dang, "Motor imagery eeg recognition based on conditional optimization empirical mode decomposition and multi-scale convolutional neural network," *Expert Systems with Applications*, vol. 149, p. 113285, 2020.

[141] Z. Tayeb, J. Fedjaev, N. Ghaboosi, C. Richter, L. Everding, X. Qu, Y. Wu, G. Cheng, and J. Conradt, "Validating deep neural networks for online decoding of motor imagery movements from eeg signals," *Sensors*, vol. 19, no. 1, p. 210, 2019.

[142] P. L. Nunez, R. Srinivasan *et al.*, *Electric fields of the brain: the neurophysics of EEG*. Oxford University Press, USA, 2006.

[143] L. Sörnmo and P. Laguna, *Bioelectrical signal processing in cardiac and neurological applications*. Academic Press, 2005, vol. 8.

[144] 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.

[145] H. Wang, T. Li, and Z. Huang, "Remote control of an electrical car with ssvep-based bci," in *2010 ieee international conference on information theory and information security*. IEEE, 2010, pp. 837–840.

[146] S. Mouli, R. Palaniappan, I. P. Sillitoe, and J. Q. Gan, "Performance analysis of multi-frequency ssvep-bci using clear and frosted colour led stimuli," in *13th IEEE International Conference on BioInformatics and BioEngineering*. IEEE, 2013, pp. 1–4.

[147] G. R. Muller-Putz and G. Pfurtscheller, "Control of an electrical prosthesis with an ssvep-based bci," *IEEE Transactions on biomedical engineering*, vol. 55, no. 1, pp. 361–364, 2007.

[148] P. F. Diez, V. A. Mut, E. M. A. Perona, and E. L. Leber, "Asynchronous bci control using high-frequency ssvep," *Journal of neuroengineering and rehabilitation*, vol. 8, no. 1, pp. 1–9, 2011.

[149] V. J. Samar, A. Bopardikar, R. Rao, and K. Swartz, "Wavelet analysis of neuroelectric waveforms: a conceptual tutorial," *Brain and language*, vol. 66, no. 1, pp. 7–60, 1999.

[150] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, and H. H. Liu, "The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis," *Proceedings of the Royal Society of London. Series A: mathematical, physical and engineering sciences*, vol. 454, no. 1971, pp. 903–995, 1998.

[151] Z. Zhang, X. Li, and Z. Deng, "A cwt-based ssvep classification method for brain-computer interface system," in *2010 International Conference on Intelligent Control and Information Processing*. IEEE, 2010, pp. 43–48.

[152] M. Kumari and S. B. Somani, "Enhancing the classification accuracy of ssvep based bci using cwt method along with ann," *International Journal of Advanced Research in Engineering and Management*, 2015.

[153] W.-Y. Hsu and Y.-N. Sun, "Eeg-based motor imagery analysis using weighted wavelet transform features," *Journal of neuroscience methods*, vol. 176, no. 2, pp. 310–318, 2009.

[154] N. E. Huang, S. R. Long, and Z. Shen, "The mechanism for frequency downshift in nonlinear wave evolution," *Advances in applied mechanics*, vol. 32, pp. 59–117C, 1996.

[155] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural networks*, vol. 13, no. 4-5, pp. 411–430, 2000.

[156] C. Guger, H. Ramoser, and G. Pfurtscheller, "Real-time eeg analysis with subject-specific spatial patterns for a brain-computer interface (bci)," *IEEE transactions on rehabilitation engineering*, vol. 8, no. 4, pp. 447–456, 2000.

[157] W. Samek, C. Vidaurre, K.-R. Müller, and M. Kawanabe, "Stationary common spatial patterns for brain–computer interfacing," *Journal of neural engineering*, vol. 9, no. 2, p. 026013, 2012.

[158] "Bci competition iii," https://www.bbci.de/competition/iii/, (Accessed on 09/04/2021).

[159] H. Cho, M. Ahn, S. Ahn, M. Kwon, and S. Jun, "Gigadb dataset - supporting data for "eeg datasets for motor imagery brain computer interface"," http://gigadb.org/dataset/100295, (Accessed on 1/2/2019).

[160] B. BCI, "Bci competition iv," http://www.bbci.de/competition/iv/, (Accessed on 1/2/2019).

[161] H. Cho, M. Ahn, S. Ahn, M. Kwon, and S. C. Jun, "Eeg datasets for motor imagery brain–computer interface," *GigaScience*, vol. 6, no. 7, p. gix034, 2017.

[162] T. Poggio, H. Mhaskar, L. Rosasco, B. Miranda, and Q. Liao, "Why and when can deep-but not shallow-networks avoid the curse of dimensionality: a review," *International Journal of Automation and Computing*, vol. 14, no. 5, pp. 503–519, 2017.

[163] G. Ruffini, "Models, networks and algorithmic complexity," *arXiv preprint arXiv:1612.05627*, 2016.

[164] J. Lee, J. Park, K. L. Kim, and J. Nam, "Sample-level deep convolutional neural networks for music auto-tagging using raw waveforms," *arXiv preprint arXiv:1703.01789*, 2017.

[165] A. A. Gunawan, K. Surya *et al.*, "Brainwave classification of visual stimuli based on low cost eeg spectrogram using densenet," *Procedia Computer Science*, vol. 135, pp. 128–139, 2018.

[166] A. Zabidi, W. Mansor, Y. Lee, and C. C. W. Fadzal, "Short-time fourier transform analysis of eeg signal generated during imagined writing," in *2012 International Conference on System Engineering and Technology (ICSET)*. IEEE, 2012, pp. 1–4.

[167] B. Boashash, *Chapter 2: Heuristic Formulation of Time-Frequency Distributions. Time-frequency signal analysis and processing: a comprehensive reference*. Academic Press, 2015.

[168] H. Nolan, R. Whelan, and R. B. Reilly, "Faster: fully automated statistical thresholding for eeg artifact rejection," *Journal of neuroscience methods*, vol. 192, no. 1, pp. 152–162, 2010.

[169] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[170] C. Schuster, R. Hilfiker, O. Amft, A. Scheidhauer, B. Andrews, J. Butler, U. Kischka, and T. Ettlin, "Best practice for motor imagery: a systematic literature review on motor imagery training elements in five different disciplines," *BMC Medicine*, vol. 9, no. 1, Jun. 2011. [Online]. Available: https://doi.org/10.1186/1741-7015-9-75

[171] L. Bozhkov, "Overview of deep learning architectures for classifying brain signals," *KNOWLEDG EE SOCIET YY*, p. 54, 2016.

[172] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4694–4702.

[173] Y. Shen, H. Lu, and J. Jia, "Classification of motor imagery eeg signals with deep learning models," in *Intelligence Science and Big Data Engineering: 7th International Conference, IScIDE 2017, Dalian, China, September 22-23, 2017, Proceedings 6*. Springer, 2017, pp. 181–190.

[174] P. Bashivan, I. Rish, M. Yeasin, and N. Codella, "Learning representations from eeg with deep recurrent-convolutional neural networks," *arXiv preprint arXiv:1511.06448*, 2015.

[175] Autonomio, "Talos optimisation tool." [Online]. Available: https://github.com/autonomio/talos

[176] S. Lemm, B. Blankertz, T. Dickhaus, and K.-R. Müller, "Introduction to machine learning for brain imaging," *Neuroimage*, vol. 56, no. 2, pp. 387–399, 2011.

[177] S. Narejo, E. Pasero, and F. Kulsoom, "Eeg based eye state classification using deep belief network and stacked autoencoder," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 6, no. 6, pp. 3131–3141, 2016.

[178] Y.-P. Lin and T.-P. Jung, "Improving eeg-based emotion classification using conditional transfer learning," *Frontiers in human neuroscience*, vol. 11, p. 334, 2017.

[179] 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.

[180] Y. R. Tabar and U. Halici, "A novel deep learning approach for classification of eeg motor imagery signals," *Journal of neural engineering*, vol. 14, no. 1, p. 016003, 2016.

[181] H. Yang, S. Sakhavi, K. K. Ang, and C. Guan, "On the use of convolutional neural networks and augmented csp features for multi-class motor imagery of eeg signals classification," in *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2015, pp. 2620–2623.

[182] A. Singh, S. Lal, and H. W. Guesgen, "Reduce calibration time in motor imagery using spatially regularized symmetric positives-definite matrices based classification," *Sensors*, vol. 19, no. 2, p. 379, 2019.

[183] M. Vansteensel, G. Kristo, E. Aarnoutse, and N. Ramsey, "The brain-computer interface researcher's questionnaire: from research to application," *Brain-Computer Interfaces*, vol. 4, no. 4, pp. 236–247, 2017.

[184] W. Speier, I. Fried, and N. Pouratian, "Improved p300 speller performance using electrocorticography, spectral features, and natural language processing," *Clinical Neurophysiology*, vol. 124, no. 7, pp. 1321–1328, 2013.

[185] N. G. Tavares and R. Gad, "Steady-state visual evoked potential-based real-time bci for smart appliance control," in *Cognitive Informatics and Soft Computing*. Springer, 2019, pp. 795–805.

[186] D. P. Bhemjibhaih, G. D. Sanjay, V. Sreejith, and B. Prakash, "Brain-computer interface based home automation system for paralysed people," in *2018 IEEE Recent Advances in Intelligent Computational Systems (RAICS)*. IEEE, 2018, pp. 230–233.

[187] D. Elstob and E. L. Secco, "A low cost eeg based bci prosthetic using motor imagery," *arXiv preprint arXiv:1603.02869*, 2016.

[188] G. R. Müller-Putz, P. Ofner, A. Schwarz, J. Pereira, G. Luzhnica, C. di Sciascio, E. Veas, S. Stein, J. Williamson, R. Murray-Smith *et al.*, "Moregrasp: Restoration of upper limb function in individuals with high spinal cord injury by multimodal neuroprostheses for interaction in daily activities," *7th Graz Brain-Computer Interface Conference 2017*, 2017.

[189] M. Gomez-Rodriguez, M. Grosse-Wentrup, J. Hill, A. Gharabaghi, B. Schölkopf, and J. Peters, "Towards brain-robot interfaces in stroke rehabilitation," in *2011 IEEE International Conference on Rehabilitation Robotics*. IEEE, 2011, pp. 1–6.

[190] R. Abiri, X. Zhao, G. Heise, Y. Jiang, and F. Abiri, "Brain computer interface for gesture control of a social robot: An offline study," in *2017 Iranian Conference on Electrical Engineering (ICEE)*. IEEE, 2017, pp. 113–117.

[191] T. P. Luu, S. Nakagome, Y. He, and J. L. Contreras-Vidal, "Real-time eeg-based brain-computer interface to a virtual avatar enhances cortical involvement in human treadmill walking," *Scientific reports*, vol. 7, no. 1, pp. 1–12, 2017.

[192] B. Kerous, F. Skola, and F. Liarokapis, "Eeg-based bci and video games: a progress report," *Virtual Reality*, vol. 22, no. 2, pp. 119–135, 2018.

[193] B. van de Laar, B. Reuderink, D. P.-O. Bos, and D. Heylen, "Evaluating user experience of actual and imagined movement in bci gaming," in *Interdisciplinary Advancements in Gaming, Simulations and Virtual Environments: Emerging Trends*. IGI Global, 2012, pp. 266–280.

[194] A. Kawala-Janik, M. Podpora, A. Gardecki, W. Czuczwara, J. Baranowski, and W. Bauer, "Game controller based on biomedical signals," in *2015 20th International Conference on Methods and Models in Automation and Robotics (MMAR)*. IEEE, 2015, pp. 934–939.

[195] B. Binias, D. Myszor, and K. A. Cyran, "A machine learning approach to the detection of pilot's reaction to unexpected events based on eeg signals," *Computational intelligence and neuroscience*, vol. 2018, 2018.

[196] A. Wadeson, A. Nijholt, and C. S. Nam, "Artistic brain-computer interfaces: state-of-the-art control mechanisms," *Brain-Computer Interfaces*, vol. 2, no. 2-3, pp. 70–75, 2015.

[197] M. Andujar and J. E. Gilbert, "Let's learn! enhancing user's engagement levels through passive brain-computer interfaces," in *CHI'13 extended abstracts on human factors in computing systems*, 2013, pp. 703–708.

[198] F. Perales and E. Amengual, "Combining eeg and serious games for attention assessment of children with cerebral palsy," in *Converging Clinical and Engineering Research on Neurorehabilitation II*. Springer, 2017, pp. 395–399.

[199] E. R. Miranda, S. Durrant, and T. Anders, "Towards brain-computer music interfaces: Progress and challenges," in *2008 First International Symposium on Applied Sciences on Biomedical and Communication Technologies*. IEEE, 2008, pp. 1–5.

[200] D. Todd, P. J. McCullagh, M. D. Mulvenna, and G. Lightbody, "Investigating the use of brain-computer interaction to facilitate creativity," in *Proceedings of the 3rd Augmented Human International Conference*, 2012, pp. 1–8.

[201] 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.

[202] R. W. Veryzer and B. Borja de Mozota, "The impact of user-oriented design on new product development: An examination of fundamental relationships," *Journal of product innovation management*, vol. 22, no. 2, pp. 128–143, 2005.

[203] A. J. Lazard, I. Watkins, M. S. Mackert, B. Xie, K. K. Stephens, and H. Shalev, "Design simplicity influences patient portal use: the role of aesthetic evaluations for technology acceptance," *Journal of the American Medical Informatics Association*, vol. 23, no. e1, pp. e157–e161, 2016.

[204] A. W. Mackelprang, M. Habermann, and M. Swink, "How firm innovativeness and unexpected product reliability failures affect profitability," *Journal of Operations Management*, vol. 38, pp. 71–86, 2015.

[205] 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.

[206] J. W. Kam, S. Griffin, A. Shen, S. Patel, H. Hinrichs, H.-J. Heinze, L. Y. Deouell, and R. T. Knight, "Systematic comparison between a wireless eeg system with dry electrodes and a wired eeg system with wet electrodes," *NeuroImage*, vol. 184, pp. 119–129, 2019.

[207] V. Mihajlovic, G. G. Molina, and J. Peuscher, "To what extent can dry and water-based eeg electrodes replace conductive gel ones?" in *proc. BIODEVICES conference, Vilamoura, Algarve, Portugal*, 2012.

[208] F. Jurysta, P. Van De Borne, P.-F. Migeotte, M. Dumont, J.-P. Lanquart, J.-P. Degaute, and P. Linkowski, "A study of the dynamic interactions between sleep eeg and heart rate variability in healthy young men," *Clinical neurophysiology*, vol. 114, no. 11, pp. 2146–2155, 2003.

[209] S.-H. Seo and J.-T. Lee, "Stress and eeg," *Convergence and hybrid information technologies*, vol. 27, 2010.

[210] B. Lee, C. Y. Liu, and M. L. Apuzzo, "A primer on brain–machine interfaces, concepts, and technology: a key element in the future of functional neurorestoration," *World neurosurgery*, vol. 79, no. 3-4, pp. 457–471, 2013.

[211] Z. C. Chao, Y. Nagasaka, and N. Fujii, "Long-term asynchronous decoding of arm motion using electrocorticographic signals in monkey," *Frontiers in neuroengineering*, vol. 3, p. 3, 2010.

[212] C. Vidaurre and B. Blankertz, "Towards a cure for bci illiteracy," *Brain topography*, vol. 23, no. 2, pp. 194–198, 2010.

[213] M. Maruishi, Y. Tanaka, H. Muranaka, T. Tsuji, Y. Ozawa, S. Imaizumi, M. Miyatani, and J. Kawahara, "Brain activation during manipulation of the myoelectric prosthetic hand: a functional magnetic resonance imaging study," *NeuroImage*, vol. 21, no. 4, pp. 1604–1611, Apr. 2004. [Online]. Available: https://doi.org/10.1016/j.neuroimage.2003.12.001

[214] G. Tian and Y. Liu, "Study on classification of left-right hands motor imagery EEG signals based on CNN," in *2018 IEEE 17th International Conference on Cognitive Informatics and Cognitive Computing*. IEEE, jul 2018. [Online]. Available: https://doi.org/10.1109/icci-cc.2018.8482042

[215] C. Molnar, *Interpretable machine learning*. Lulu. com, 2020.

[216] M. A. Ahmad, C. Eckert, and A. Teredesai, "Interpretable machine learning in healthcare," in *Proceedings of the 2018 ACM international conference on bioinformatics, computational biology, and health informatics*, 2018, pp. 559–560.

[217] M. Abadi and A. A. et al., "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[218] R. Al-Rfou, G. Alain, A. Almahairi, C. Angermueller, D. Bahdanau, N. Ballas, F. Bastien, J. Bayer, A. Belikov, A. Belopolsky *et al.*, "Theano: A python framework for fast computation of mathematical expressions," *arXiv e-prints*, pp. arXiv–1605, 2016.

[219] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alche-Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 8024–8035. [Online]. Available: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

[220] F. Chollet *et al.* (2015) Keras. [Online]. Available: https://github.com/fchollet/keras

[221] "Autonomio talos [computer software]. (2020). retrieved from http://github.com/autonomio/talos." https://github.com/autonomio/talos, (Accessed on 10/2/2021).

[222] R. Caruana, S. Lawrence, and C. L. Giles, "Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping," in *Advances in neural information processing systems*, 2001, pp. 402–408.

[223] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting." *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014. [Online]. Available: http://www.cs.toronto.edu/~rsalakhu/papers/srivastava14a.pdf

[224] L. Prechelt, "Early stopping-but when?" in *Neural Networks: Tricks of the trade*. Springer, 1998, pp. 55–69.

[225] C. Darken, J. Chang, and J. Moody, "Learning rate schedules for faster stochastic gradient search," in *Neural Networks for Signal Processing II Proceedings of the 1992 IEEE Workshop*. IEEE, 2002. [Online]. Available: https://doi.org/10.1109/nnsp.1992.253713

[226] Y. N. Dauphin, R. Pascanu, C. Gulcehre, K. Cho, S. Ganguli, and Y. Bengio, "Identifying and attacking the saddle point problem in high-dimensional non-convex optimization," in *Advances in neural information processing systems*, 2014, pp. 2933–2941.

[227] L. Merrick and Q. Gu, "Exploring the use of adaptive gradient methods in effective deep learning systems," in *2018 Systems and Information Engineering Design Symposium (SIEDS)*. IEEE, apr 2018. [Online]. Available: https://doi.org/10.1109/sieds.2018.8374740

[228] "Cs231n convolutional neural networks for visual recognition," https://cs231n.github.io/neural-networks-3/, (Accessed on 02/11/2021).

[229] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2121–2159, 2011.

[230] M. D. Zeiler, "Adadelta: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[231] T. Tieleman and G. Hinton, "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude," *COURSERA: Neural networks for machine learning*, vol. 4, no. 2, pp. 26–31, 2012.

[232] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[233] T. Dozat, "Incorporating nesterov momentum into adam," *International Conference on Learning Representations*, 2016.

[234] S. J. Reddi, S. Kale, and S. Kumar, "On the convergence of adam and beyond," *International Conference on Learning Representations*, 2018.

[235] F. Neukart, S.-A. Moraru, C.-M. Grigorescu, and P. Szakacs-Simon, "Cortical artificial neural networks and their evolution, consciousness-inspired data mining," in *2012 13th International Conference on Optimization of Electrical and Electronic Equipment (OPTIM)*. IEEE, may 2012. [Online]. Available: https://doi.org/10.1109/optim.2012.6231782

[236] A. Neelakantan, L. Vilnis, Q. V. Le, I. Sutskever, L. Kaiser, K. Kurach, and J. Martens, "Adding gradient noise improves learning for very deep networks," *arXiv preprint arXiv:1511.06807*, 2015.

[237] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research*, vol. 13, no. Feb, pp. 281–305, 2012.

Appendices

Appendix A

EEG Signal Artefacts

A.1 Physiological Artefacts

There are several physiological artefacts including ocular distortion, muscle activity and cardiac activity, respiration, and perspiration. In the next few paragraphs these artefacts are discussed.

Ocular distortion is known as the EOG, this occurs during blinking, lateral movement and general eye movements. When electrically modelled, the eye may be represented by a dipole. It distorts the electric field in the region when the eye moves. EOG signals have a magnitude larger than that of EEG (reaching around 100-200 mV). Blinking produces quick changes in the electrodes places at the frontal area of the brain, which is more concentrated whilst closer to the eyes. Lateral movements affect the frontal region but also distort EEG signals near the temples. Some ocular activity man be confused with delta and theta bands in low frequencies. The resulting signal created by electrical activity in the muscles is called electromyography. The signals interfere with EEG, which can be observed by the naked eye as high–frequency artefacts. Movements such as clenching the jaw, chewing, talking, frowning and swallowing all contribute to the signal distortion, the amplitude of which is correlated directly to the strength of the muscle contraction. In particular, these affect the high frequency (beta and gamma) EEG bands.

Cardiac artefacts are produced by electrical activity from the heart. The signals are called Electrocardiogram. Though the signal is of low amplitude on the scalp, depending on where the electrode has been placed, it is

possible to see a rhythmic distortion in the resulting EEG. ECG and EEG have overlapping frequency bands and thus are difficult to visualise.

Sweat glands produce perspiration in the skin. A small amount of sweat produced may cause a change in the electrical baseline of the electrodes. With a higher amount of sweat, there is a possibility of creating shorts between electrodes. On the resulting EEG, these can be shown as slow waved, which overlap the EEG signal. These can be shown in the frequency domain as a low–frequency artefact that generally overlaps the delta and theta bands.

Respiration is normally an artefact when recording subjects whilst they are sleeping. The electrodes' contact to the scalp may modify depending on the movement of the head and chest whilst breathing. The resulting artefact can be seen on the EEG as slow waves, which are synchronised with the subject's breathing. Respiration artefacts are of low frequency and overlap in the delta and theta bands.

A.2 Non–Physiological Artefacts

The presence of non-physiological artefacts in EEG recordings can severely affect the validity of the results and make it challenging to extract meaningful information. Some of the most common types of non-physiological artefacts include electrode pop, cable movement, incorrect reference placement, AC electrical and electromagnetic interferences, and body movement.

Electrode pop is produced due to temporary failure in contact between the EEG electrode and the scalp or by touching the sensor. The resulting signal is abrupt and usually of high amplitude, and generally localised in a signal channel. The artefact could also be produced due to the movement of the cables that connect the electrodes and the amplification system. Changes are produced in the electromagnetic fields, which distort the signal recorded and possibly the sensor contact to the scalp. In addition to this, correct placement and contact of the reference sensor is vital. In the case where the placement or contact is not sufficient, the EEG signal is not recorded.

Abrupt changes will be seen on all channels, which are of high amplitude. One can observe high power in all channels (EEG and Non–EEG related signals) in the frequency domain. AC interference from electrical lines is produced either at 50Hz or 60Hz, depending on the recording country. If there is insufficient wire shielding, the signal can be affected by the surrounding electromagnetic fields. A continuously overlapping high–frequency noise is observed on the EEG signal. In the frequency domain, a large spike is observed at 50Hz or 60Hz.

Finally, body movements, even if unintentional, the contact between the electrode and the skin may be affected, thus corrupting the EEG signal. Any body movement can distort EEG signals; however, head movements are the principal cause. These are observed in the EEG as temporary slow waves which correspond to the rhythm of movement. This is seen typically overlapping in the lower frequencies (delta and theta bands) in the frequency domain.

It is crucial to minimise the impact of these non-physiological artefacts on EEG recordings in order to obtain accurate and reliable results. To achieve this, appropriate technical procedures and precautions should be taken during the recording process.

Appendix B

Experimental Setup

All of the data analysis conducted during the course of this research was conducted on the EVGA GeForce GTX 1070 Ti GPU, which runs the NVIDIA Pascal programming language.

This set of experiments was built on top of the open-source TensorFlow machine learning framework [217]. According to the researchers, when TensorFlow was used alone, a tedious process with a steep learning curve was discovered. TensorFlow's popularity is reasonably clear, with many large companies utilising TensorFlow to perform artificial intelligence activities. Many well-known companies, including NVIDIA, Twitter, Snapchat, and Uber, rely on TensorFlow for their major operations and research. On the one side, one could argue that TensorFlow's appeal stems from its history and legacy. TensorFlow is a household name due to its development under the aegis of "Google". Without question, TensorFlow has received more marketing attention than some of its competitors such as Theano [218], and PyTorch [219].

Kernel-level API built on TensorFlow, which provides a more intuitive set of abstractions that simplifies the development of deep learning models. Keras [220] is a open-source deep learning library which works on top of Google's Tensorflow framework. The models presented in this thesis were created using the Keras libraries. Its ability to make machine learning accessible to both industry and academic researchers aided in its rise to prominence as the most popular deep learning platform. Keras adheres to best practices for minimising cognitive load by providing consistent and straightforward APIs, minimising the number of user activities required for typical use cases, and providing clear and responsive feedback in the event of user error.

The Talos library [221] was used to optimise hyperparameters because manually tuning them via empirical

observations is a time-consuming and difficult task. Talos is a model performance evaluation and tuning tool that automates the process of tuning hyperparameters. It includes optimizations for Grid Search and Random Search, as well as their implementations. In the course of this research both grid search and random search were used. It then performs subsequent model evaluations by exhaustively retraining the model using different permutations of the input hyperparameters.

Appendix C

Deep Learning Optimisation

Optimisation is required in deep learning as a method of improving how an algorithm models against a dataset. They tie in the loss function and the model parameters by changing the model as response to how the loss function is performing. Optimisers essentially shape the model to be as accurate as possible, using the loss function as a guide, telling it whether it is moving in the correct or wrong direction. Hyperparameters govern the training process itself. Within the machine learning space, hyperparameters are pre-initialised variables defined before training takes place. These can be modified in order to alter the behavior of the algorithm, to optimise.

The Cost Functions is a core reason networks require evaluation and optimisation. Simply, if the loss function outputs a high number, the model is not working well; however if it output a low loss function, it shows a better model, outputting a low number. Loss functions show us how the model is able to fit the data in the first place. The two main loss functions mentioned in the latest literature are Mean Squared Error, and Cross–entropy loss.

Mean–squared Error (MSE) is a basic loss function, as the name suggests it measure of the averages of squared difference between predictions and actual observations, and is only concerned with the average magnitude of error irrespective of its direction. It is easy to understand and implement. In order to calculate MSE, the difference between the predictions and the ground truth are calculated, squared and averaged across the whole dataset. In equation form this can be described as C.1. Where Y_i is a vector which represents the true values, and $\hat{Y}_i$ denotes the values of n number of predictions.

$$MSE = J(\theta) = \frac{1}{n}\sum_{i-1}^{n}(Y_i - \hat{Y}_i)^2 \tag{C.1}$$

Cross Entropy loss is also known as the Log Loss function. Cross–entropy measures the divergence between two probability distributions. If the cross–entropy is large, i.e. the distribution between the two is large, and vise–versa. Essentially the log of the actual predicted value probability is being multiplied with the ground truth class.The formula for this is much like the likelihood loss functions, with added algorithms to it, as seen in Equation C.2.

$$-\sum\{y_i \ln(\hat{Y}_i) + (1 - Y_i)log(1 - \hat{Y}i)\} \tag{C.2}$$

C.1 Regularisation

Regularisation reduces the complexity of the model in order to discourage the model learning a complex model which would have a higher risk of overfitting. The more training data there is, the possibility of overfitting can be expensive for the model. Even if more training data is added, the generalisation accuracy may only rise by a small amount.

The primary reason why overfitting occurs is due to the network having the ability to recognise even the smallest details present in the data [222]. Due to this the model tends to perform extremely well on the training data samples, but on the validation sample results will be poor. By doing this, the model will not learn much from these units, and making the network simpler.

l_2 Regularisation is also called Ridge Regression, which adds a square magnitude coefficient as a penalty term to the loss function.l_2 does not offer feature selection, and it is not robust to outliers within data. Due to this l_2 gives better predictions when the output variable is a function of all the input features. It is preferred over l_1 regularisation due to its ability to learn more complex patterns in data.The l_2 regularisation element is defined in Eq. C.3. Added to the cost function this now becomes Eq. C.4. In this equation θ represents the weights, x represents the bias, $h_\theta x$ represents the cumulative. Here the parameter λ is the variable which we have the ability to tweak within the equation during implementation.

l_2 Regularisation element:

$$\lambda \sum_{i=1}^{n} \theta_i^2 \tag{C.3}$$

Loss Function with l_2 Regularisation term:

$$L(x, y) \equiv \sum_{i=1}^{n} (y_i - h_\theta(x_i))^2 + \lambda \sum_{i=1}^{n} \theta_i^2 \tag{C.4}$$

l_1 Regularisation is also known as Lasso Regression, and adds the absolute value of the magnitude of coefficient as a penalty term to the cost function. l_1 regularisation does feature selection by assigned insignificant input features with zero weight, and important features with a weight which is not zero – hence it is robust to any outliers.The l_1 element is described in Eq C.5. The cost function with l_1 is then re-defined as Eq. C.6. The drawback to l_1 is that the model generated are simple and interpretable although it is not able to learn complex patterns.In this equation θ represents the weights, x represents the bias, $h_\theta x$ represents the cumulative. Here the parameter λ is the variable which we have the ability to tweak within the equation during implementation.

l_1 Regularisation element:

$$\lambda \sum_{i=1}^{n} |\theta_i| \tag{C.5}$$

Loss Function with l_1 Regularisation term:

$$L(x, y) \equiv \sum_{i=1}^{n} (y_i - h_\theta(x_i))^2 + \lambda \sum_{i=1}^{n} |\theta_i| \tag{C.6}$$

Dropout is another type of regularisation technique, first introduced by Srivastava et al. [223]. Dropout is simply a method where random hidden units are turned off, at a particular probability, such as Fig. C.2; this creates a sparser network. Thus these switched off neurones would not contribute any information during the updates. This is so that every redundant detail in a training dataset is not learnt by the network, and is encouraged to learn a sparse representation as a side–effect of this.

For example, if we have a network with three hidden layers, Fig C.1. We set the dropout probability of to be 0.5 of the network, as seen in Fig C.2, i.e. 50% of the hidden units of each of these layers would be randomly removed, leaving us with a much simpler network to train.

Learning rate, usually mathematically represented by α, is parameter which allows us to control how much

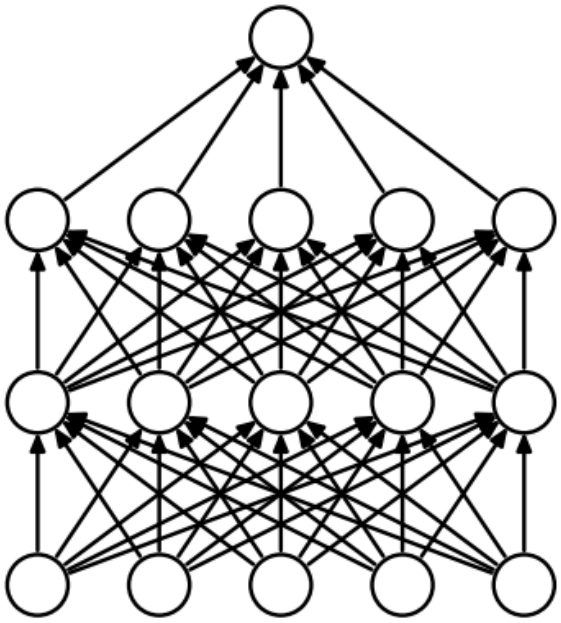


Figure C.1: Fully connected complex network [223]

the weights are being adjusted with respect to the loss gradient. Generally, a large learning rate allows the model to converge to the minimum faster, however the cost of this may be that it may never land on the true minimum. However if we take a small learning rate, though it may take a lengthier time to train, the chances of it converging to the true minimum are higher. There is a middle ground to this – decaying the learning rate over a period of time. At the beginning of training the learning rate would be at its peak, and over time, it would decay.

In practice, it was found that step decay is the most preferable of these hyperparameters. Same mean and variance to normalize the test set as training data - we want the transformation to occur both on the train and test data. The scales of features for unnormalised stat will vary, and hence there will be a variation in the parameters which are learnt for each feature. This would cause the cost function to become asymmetric. Once normalised, the cost function will be symmetric, and thus easier for gradient descent algorithm to find the global minima faster, therefore positively affecting the algorithm speed.

Early Stopping is a another form of regularisation techniques used in order to avoid overfitting during training. It is a widely used method, and is simple to understand and implement, in comparison to other regularisation techniques, [224]. The data which is split into training and validation sets are evaluated at the end of each epoch, or *Nth* epoch, in terms of its performance, and a check to see if the network has outperformed that previous best model, if so this would become the now current best model. If the network does not outperform,

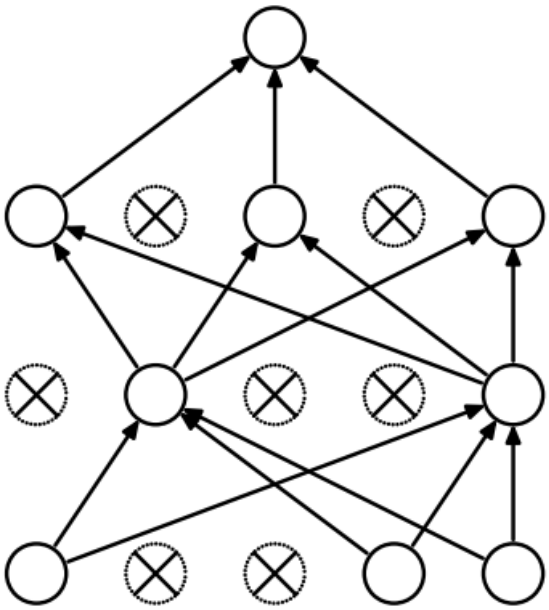


Figure C.2: Fully connected network, with 50% of each layer randomly chosen to dropout [223]

after a preset number of epochs, the training would terminate.

C.2 Gradient Descent Variants

Gradient descent and its several variants are used whilst training deep networks. It is one of the most popular training algorithms used to perform optimisation. This section describes the details of each of these variants, their benefits, and challenges.

Normal gradient descent computes the gradient of the cost function J with respect to the parameter θ for the entire training set. The mathematical representation is described by Equation C.7. The main advantage are they can be used fixed learning rate with use of learning rate decay methods. They also have unbiased estimates of gradients. The more examples, the lower the standard error. Direct trajectory towards the minimum and it theory will always converge. The main disadvantages are that it is slow to go over all examples, and will take a longer time training for larger datasets. Additionally, each learning step occurs once all the samples have been gone over, though some samples may not contribute, and thus be redundant

$$\theta = \theta - \alpha \cdot \nabla_\theta J(\theta) \tag{C.7}$$

Mini–Batch Gradient descent is much like a hybrid between normal Gradient Descent and SGD, taking the best parts of both optimisation algorithms. It performs an update for every mini–batch n of training samples, which are shuffled during training. The mathematical representation is shown by Eq. C.8. The main advantages are that this is faster than Batch Gradient Descent. When batch size is less than the size of the training set, it adds noise to the learning process, which adds to the improvement of generalisation error. Mini batch Gradient randomly selects samples which help to avoid redundant examples which do not contribute to the learning process. Disadvantages include that it will not converge due to noise, therefore would wander the minimum region and due to the noise incurred, learning–decay must be added to help to come closer to the minimum.

$$\theta = \theta - \alpha \cdot \nabla_\theta J(\theta; x^{(i:i+n)}; y^{(i:i+n)}) \tag{C.8}$$

Stochastic Gradient Descent (SGD), in contact performs parameter update for each of the training sample x^i and its label y^i, thus learning happens on every sample. Much like the Mini–batch gradient descent, the training data is shuffled to avoid pre–existing order of samples. The mathematical representation is shown in Eq.C.9. SGD shares most of its pros and cons with Mini–batch. The disadvantages specifically relate to SGD include that it adds more noise than Mini–batch, however improves generalisation error, though also increasing training time and also it cannot vectorise over one sample as training becomes very slow, and

variance would be large.

$$\theta = \theta - \alpha \cdot \nabla_\theta J(\theta; x^{(i)}; y^{(i)}) \tag{C.9}$$

There are various challenges which are faced with gradient descent strategies.

Selection of an appropriate learning rate can be difficult. If a learning rate which is too small is chosen it can be incredibly slow to converge, too high, and it may hamper the process of convergence. This could lead to the loss function fluctuation near the minimum or may even cause it to diverge. The learning rate schedules can be added to try to optimise the algorithm, by reducing the learning rate which is predefined, or between epochs which fall between a particular threshold. However these have to be defined prior to training, and thus are unable to adapt to the dataset's characteristics [225].Another key challenge is to minimise getting trapped in the ravines of highly complex error functions within their numerous local minima. It is however argued, by [226], that it is not the local minima which is the problem however, but that it the saddle points, i.e. the points where one dimension slopes up and another slopes downward. These points are generally surrounded by a plateau which incur the same error make it difficult for SGD to escape as the gradient at these point being close to zero in all dimensions.

C.3 Gradient Descent Optimisation Algorithms

SGD has trouble navigating where the surface curves are more steep in one dimension than another – this is common in the local optima. Due to this shortcoming, SGD makes hesitant progress through slopes towards the local optimum. Momentum is a modification which adds direction and dampens the oscillation of SGD, thus gain faster convergence.

Added momentum is preferred with SGD for two reasons: SGD does not compute the derivative of the loss function exactly, and thus does not always travel in the optimal direction because of these estimated derivatives, thus exponentially weighted averages would be closed to the actual derivative. The second reason is due to ravines which are common near local minima in deep learning, if the ravine is narrow, SGD has trouble with navigation, however when combined with momentum, it helps accelerate gradients in the correct direction.

The fraction, denoted by γ of the update vector of the past time step, and adds this to the current update vector, as observed in Equation C.10.

$$\begin{aligned} v_t &= \gamma v_{t-1} + \alpha \nabla_\theta J(\theta) \\ \theta &= \theta - v_t \end{aligned} \tag{C.10}$$

Nesterov Momentum is a modified version of the momentum update, and has recently become more popular, [227]. The difference can be seen in Figure C.3, here it can be seen that the lookahead gradient step is different to that of the normal momentum update.

$$\begin{aligned} v_t &= \gamma v_{t-1} + \alpha \nabla_\theta J(\theta - \gamma v_{t-1}) \\ \theta &= \theta - v_t \end{aligned} \tag{C.11}$$

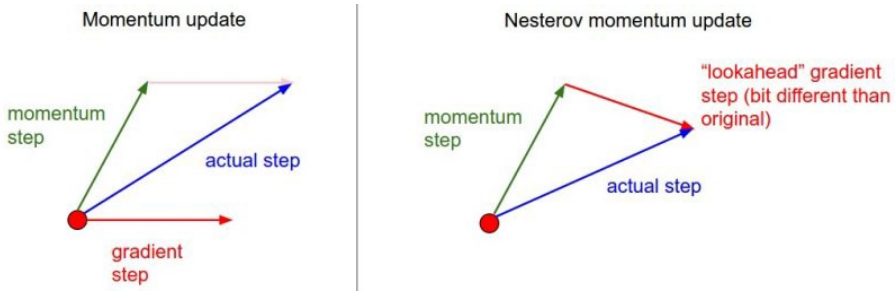


Figure C.3: Momentum update v/s Nesterov momentum update [228]

Adagrad, first introduced by [229], is an algorithm for gradient–based optimisation which adapts the learning rate to the parameters. It performs smaller updates for parameters with frequently occurring features, and larger updates for parameters associated with less frequent features. It is best suited for dealing with sparse data.

In the above equation (Equation C.11), we performed a momentum update for all parameters θ because each parameter θ_i used the same learning rate α. Equation C.12 shows us Adagrad's per parameter update, which is then vectorised: g_t is used to denote the gradient with respect to time step t, $g_{t,i}$ is the partial derivative of the objective function w.r.t to the parameter update θ_t.

$$g_{t,i} = \nabla_\theta J(\theta_{t,i}) \tag{C.12}$$

Equation C.13, shows us the SGD update for every parameter θ_i at every time step t.

$$\theta_{t+1,i} = \theta_{t,i} - \alpha \cdot g_{t,i} \tag{C.13}$$

The update rule, C.13, shows us the modification which Adagrad makes to the general learning rate α, at each time step t, for each parameter θ_i, based on its past gradients.

$$\theta_{t+1,i} = \theta_{t,i} - \frac{\alpha}{\sqrt{G_{t,ii} + \epsilon}} \cdot g_{t,i} \tag{C.14}$$

Here in Equation C.14 $G_{t,ii}$, is the diagonal matrix where each diagonal element i, i is the sum of the square of the gradients w.r.t. θi, up to the time step t.

$$\theta_{t+1,i} = \theta_{t,i} - \frac{\alpha}{\sqrt{G_{t,ii} + \epsilon}} \odot g_{t,i} \tag{C.15}$$

The implementation is now vectorised in C.15, by finding the vectorised–matrix $\odot$ of G_t and g_t.

The main benefit of Adagrad is that it takes away the the need to manually tune the learning rate. However, the downfall is that due to the accumulation of the square gradients in the denominator, over time the accumulated sum keeps growing, hence causing the learning rate to shrink and eventually become infinitesimally small to a point where the algorithm is no longer able to learn.

Adadelta, [230], is a subset of Adagrad, and it seeks to reduce the aggressive reduction in the learning rate caused by the squared gradient. Is is done by restricting the window of accumulated past gradients to a fixed size. Rather than storing previous squared gradients, it is sum of the decaying average of all past gradients.

The running average, defined as $E[g^2]_t$, at time step t, depends on the fraction γ, which is similar to the momentum term, and depends on the previous average and the current gradient C.16. Here, γ is set to a similar value to which we would set the momentum term.

$$E[g^2]_t = \gamma E[g^2]_{t-1} + (1 - \gamma) g_t^t \tag{C.16}$$

For clarity, in Equation C.17, the simple SGD parameter update is shown as the parameter update $\Delta\theta_t$

$$\begin{aligned} \Delta\theta_t &= -\alpha \cdot g_{t,i} \\ \theta_{t+1} &= \theta_t + \Delta\theta_t \end{aligned} \tag{C.17}$$

The parameter update vector is taken from Adagrad, which was previously derived, resulting in Equation C.18.

$$\Delta\theta_t = -\frac{\alpha}{\sqrt{G_{t,ii} + \epsilon}} \odot g_t \tag{C.18}$$

The diagonal matrix G_t is now replaced with the decaying average over the past squared gradients, $E[g^2]_t$.

$$\Delta\theta_t = -\frac{\alpha}{\sqrt{E[g^2]_t + \epsilon}} g_t \tag{C.19}$$

Now, in the denominator, all that remains is the root–mean squared, which can be replaced with RMS shorthand, as shown in equation C.20.

$$\Delta\theta_t = -\frac{\alpha}{RMS[g]_t} \tag{C.20}$$

We can see from the above that there is no requirement to set a default learning rate, as it has been eliminated from the update rule.

RMSProp is a gradient based optimisation technique which was first introduced by Geoffrey Hinton, [231], and is based off of SGD. It applies a clever technique in order to solve the vanishing or exploding gradient of more complex functions. It uses a moving average of squared gradients in order to normalise the gradient itself – i.e. it decreases the step size for a large gradient to avoid exploding, and increases the step size to avoid vanishing gradients.

Both RMSProp and Adadelta were both created in order to resolve the diminishing gradient issue with Adagrad. They both have an identical update vector, C.21. However for RMSProp, the suggested γ value is 0.9, and learning rate, α at 0.001.

$$
\begin{aligned}
E[g^2]_t &= 0.9E[g^2]_{t-1} + 0.1G_t^2 \\
\theta_{t+1} &= \theta_t - \frac{\alpha}{\sqrt{E[g^t]_t + \epsilon}}
\end{aligned}
\tag{C.21}
$$

Adaptive Moment Estimation, Adam, is another method which computes adaptive learning rates for each parameter, much like Adadelta and RMSProp. It stores the exponentially decayed averages of past squares, v_t but it also exponentially decays the past gradients m_t, similar to momentum. These are computed as observed in Equations C.22. v_t and m_t as the estimates of the first moment, i.e. the mean, and the second moment, i.e. the uncentered variance, giving Adam its name. Since v_t and m_t are both initially set to vectors of 0's, it was observed that they are also initially biased towards 0's, especially during the initial time–steps, and also when the decay rates are seen to be small, i.e β_1 and β_2 are close to 1. Here β_1 represents the exponential decay rate for the first moment estimates, and β_2 represents the decay rate for the second moment estimates

$$
\begin{aligned}
m_t &= \beta_1 m_{t-1} + (1-\beta_1)g_t \\
v_t &= \beta_2 v_{t-1} + (1-\beta_2)g_t^2
\end{aligned}
\tag{C.22}
$$

In order to counteract these possible biases, by computing bias–corrected first and second moment estimates, show in Equation C.23.

$$
\begin{aligned}
\hat{m}_t &= \frac{m_t}{1-\beta_1^t} \\
\hat{v}_t &= \frac{v_t}{1-\beta_2^t}
\end{aligned}
\tag{C.23}
$$

$\hat{m}_t$ and $\hat{v}_t$ are then used to update the parameters just as we have seen in Adadelta and RMSProp, given in the Adam update rule, C.24

$$
\theta_{t+1} = \theta_t - \frac{\alpha}{\sqrt{\hat{v}_t} + \epsilon}\hat{m}_t \tag{C.24}
$$

It was suggested by the authors of Adam to set default values as follows: β_1, 0.9, β_2, 0.999 and ϵ, 10^{-8}.

Adam was first introduced by Kingma et al. [232].The v_t factor in the Adam update rule scales the gradient to

be inversely proportional to the l_2 norm of past gradient, v_{t-1} and the current gradient $|g_t|^2$, as it was seen in C.22. This update can be generalised to the l_p norm, and also parametrise β_2 to be β_2^p, C.25.

$$v_t = \beta_2^p v_{t-1} + (1 - \beta_2^p)|g_t|^p \tag{C.25}$$

Norms for large values of p generally become numerically unstable – this is why l_1 and l_2. However it was found that l_∞ generally also converges to a stable value. In order to avoid confusion with Adam, u_t is used to denote the infinity constrained norm v_t, C.26.

$$\begin{aligned} u_t &= \beta_2^\infty v_{t-1} + (1 - \beta_2^\infty)|g_t|^\infty \\ &= \max(\beta_2 \cdot v_{t-1}, |g_t|) \end{aligned} \tag{C.26}$$

This is then plugged into the Adam update rule, and $\sqrt{\hat{v}_t} + \epsilon$ is replace with u_t, giving us the AdaMax update rule to be C.27

$$\theta_{t+1} = \theta_t - \frac{\alpha}{u_t}\hat{m}_t \tag{C.27}$$

Nesterov-accelerated Adaptive Moment Estimation is a combination of Adam and Nesterov accelerated gradient. In order to include Nesterov accelerated gradient into Adam, we first need to modify its momentum term m_t.

The update rule using the current notation is as follows, in equation C.28, where J is the objective function, γ is the momentum decay and α is the step size. The expansion of these terms gives us the equation C.29, which shows us that the momentum here involves us taking a step in the direction of the previous movement vector, and following this takes a step in the current gradient.

$$\begin{aligned} g_t &= \nabla_{\theta_t} J(\theta_t) \\ m_t &= \gamma m_{t-1} + \alpha g_t \\ \theta_{t+1} &= \theta_t - m_t \end{aligned} \tag{C.28}$$

$$\theta_{t+1} = \theta_t - (\gamma m_{t-1} + \alpha g_t) \tag{C.29}$$

It was proposed by Dozat [233], to rather than updating the gradient g_t and the parameters separately θ_{t+1}, the look–ahead momentum vector is updated directly to the current parameters in C.30.

$$\begin{aligned} g_t &= \nabla_{\theta_t} J(\theta_t - \gamma m_{t-1}) \\ m_t &= \gamma m_{t-1} + \alpha g_t \\ \theta_{t+1} &= \theta_t - m_t \end{aligned} \tag{C.30}$$

Here in C.31, the current momentum vector is now used. In order to add the Nesterov momentum to Adam, the previous momentum vector is also replaced with the current one.

$$\begin{aligned} g_t &= \nabla_{\theta_t} J(\theta_t) \\ m_t &= \gamma m_{t-1} + \alpha g_t \\ \theta_{t+1} &= \theta_t - (\gamma m_t + \alpha g_t) \end{aligned} \tag{C.31}$$

The Adam update rule was as follows, C.32.

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1 - \beta_1) g_t \\ \hat{m}_t &= \frac{m_t}{1 - \beta_1^t} \\ \theta_{t+1} &= \theta_t - \frac{\alpha}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t \end{aligned} \tag{C.32}$$

It is then expanded, using the definitions of $\hat{m}_t$ and m_t in C.32.

$$\theta_{t+1} = \theta_t - \frac{\alpha}{\sqrt{\hat{v}_t} + \epsilon} \left(\frac{\beta_1 m_{t-1}}{1 - \beta_1^t} + \frac{(1 - \beta_1) g_t}{1 - \beta_1^t} \right) \tag{C.33}$$

Here, $\frac{(1 - \beta_1) g_t}{1 - \beta_1^t}$ is the bias–corrected estimate of the momentum of the previous time step, and hence can be

replaced with $\hat{m}_t - 1$.

$$\theta_{t+1} = \theta_t - \frac{\alpha}{\sqrt{\hat{v}_t} + \epsilon}(\beta_1 \hat{m}_{t-1} + \frac{(1-\beta_1)g_t}{1-\beta_1^t}) \tag{C.34}$$

In order to give us the Nadam update rule, Equation C.35, $\hat{m}_{t-1}$, the previous time step, is replaced with the current momentum vector $\hat{m}_t$.

$$\theta_{t+1} = \theta_t - \frac{\alpha}{\sqrt{\hat{v}_t} + \epsilon}(\beta_1 \hat{m}_t + \frac{(1-\beta_1)g_t}{1-\beta_1^t}) \tag{C.35}$$

AMSGrad, was first introduced in 2018 by Reddi et al, [234], and is the newest of the Optimsation algorithms. They pinpointed that the square of past gradients was a reason for the poor generalisation behaviour of adaptive learning rate methods due to the short term memory which sometimes shows as a disadvantage. In order to fix this issue, AMSGrad was proposed. Here they suggested to find the maximum of the past squared gradients, v_t, rather than using the exponential average in order to update the parameters. This results in a non–increasing step–size, avoiding the issue which is faced by Adam. It is defined the same in Adam, C.36.

$$v_t = \beta_2 v_{t-1} + (1-\beta_2)g_t^2 \tag{C.36}$$

Now rather than using v_t or the bias–corrected version $\hat{v}_t$, the previous, v_{t-1} is employed if it is larger than the current value, C.37.

$$\hat{v}_t = max(\hat{v}_{t-1}, v_t) \tag{C.37}$$

Thus results in a non-increasing step–size, avoiding the issue which Adam faces. For simplicity the de-biasing step is removed. AMSGrad updated without bias–corrected parameter updates is shown as follows in C.38.

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1-\beta_1)g_t \\ v_t &= \beta_2 v_{t-1} + (1-\beta_2)g_t^2 \\ \hat{v}_t &= max(\hat{v}_{t-1}, v_t) \\ \theta_{t+1} &= \theta_t - \frac{\alpha}{\sqrt{\hat{v}_t} + \epsilon} m_t \end{aligned} \tag{C.38}$$

C.4 Network Structure

Activation functions are a quintessentially important feature of artificial neural networks. Without activation functions, the weights and bias would do a linear transformation which is easily solved, and hence does not have the capacity to address more complex problems, such as image classification, and predictions algorithms. They decide whether a neuron should be activated or not during backpropagation, and makes this decision based on the weight and bias information which it receives and applies these to the weights. Activation functions are also crucial for squashing the output of the network to be within certain bounds. This is particularly important when the outputs of the layer are very large values, where this will have a knock–on effect where subsequent layers may produce even larger values, making the process computationally intractable.

$$Y = Activation(\sum(weight \times input) + bias) \tag{C.39}$$

Based on the equation C.39, the transformed output would be the input for the next layer of the network.

The binary function (Figure C.4) is the simplest of the activation functions, and is used in a binary classifier. This is typically not used in practical applications. However, the gradient of this function is 0, and hence would not backpropagate, and update weights. Hence there is no model improvement.

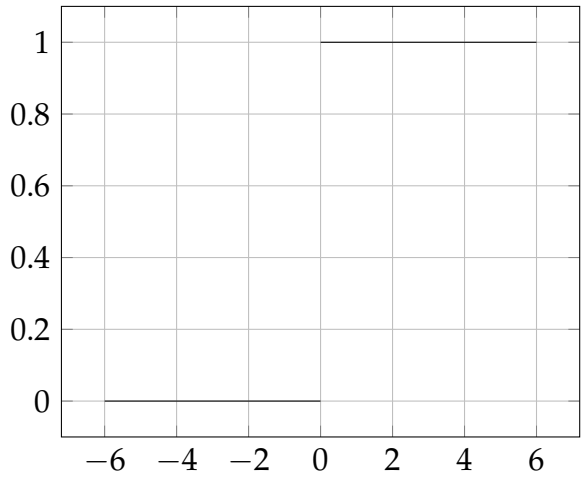


Figure C.4: Binary Step Function

$$f(x) = 1, x >= 0 \tag{C.40}$$

The linear activation function (Figure C.5) allows inputs to pass it without applying any modification. The function is defined as below, C.41. When applying backpropagation, the gradient is constant, C.42. For simple tasks, linear activation functions would be useful, however due to the constant gradient, where multiple layers would be required in the network, the output would only be a linear transformation of the input, and would not be desirable for a complex task.

$$f(x) = ax \tag{C.41}$$

$$f'(x) = a \tag{C.42}$$

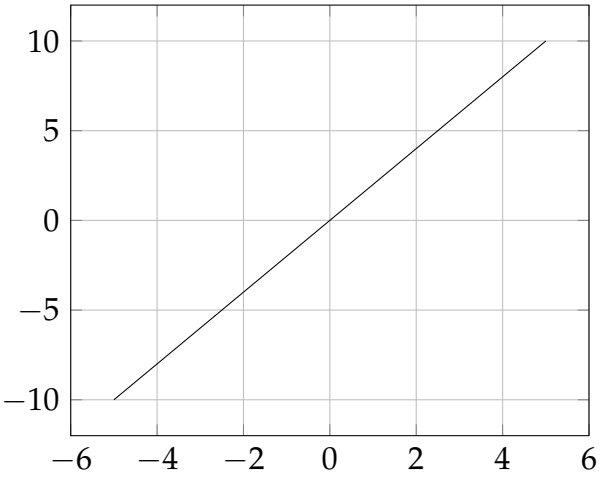


Figure C.5: Example of a Linear Activation Function: $y = 2x$

The Sigmoid Function (Figure C.6) has the ability to be constantly differentiable. The error is backpropagated, and weights are updated appropriately. This function is typically used for classification problems, and popularly used in the last layer of the network. However the sigmoid activation function has three main drawbacks. Firstly, the vanishing gradient issue is commonplace with sigmoid activation function. They have near zero gradients both when near 0 or near 1. During backward propagation through the network, some neurons may become saturated i.e. near–zero gradients, and any weights which are subsequently connected to such neurons are slowly updated as well. With more neurons becoming saturated, the network may not have the ability to backpropagate. Secondly, the outputs of sigmoid functions are not zero-centred and finally, sigmoid functions are computationally expensive due to the use of the exponential function as we can see in C.43.

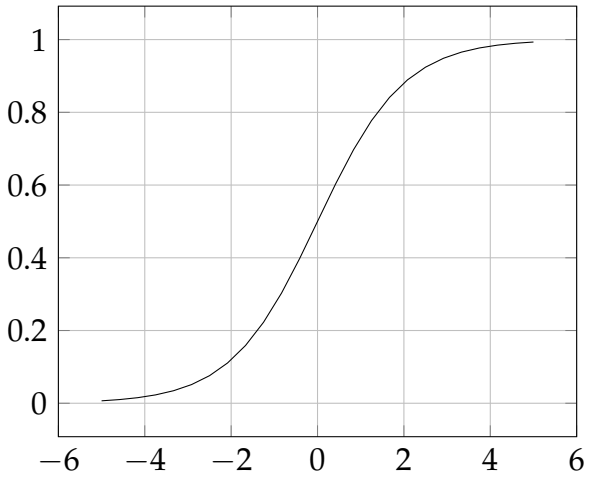


Figure C.6: Sigmoid Activation Function

$$f(x) = \frac{1}{(1 + e^{-x})} \tag{C.43}$$

The tanh activation function, also known as the hyperbolic tangent function (Figure C.7), is a scaled version of the sigmoid. Importantly, it addresses the zero-centred issue which the sigmoid activation function has. The outputs of the tanh function are between -1 and 1, which is unlike the sigmoid function where the outputs are between 0 and 1.

The gradient of the tanh function is steeper in comparison to the sigmoid. Choosing between the two functions depends on the gradient which is required by the problem in question. However, like the sigmoid function it has the drawback of vanishing gradients.

$$tanh(x) = \frac{2}{(1 + e^{-2x})} - 1 \tag{C.44}$$

The softmax activation layer is a type of sigmoid function, and is also ideally usesd as the output layer of a classification network. However unlike the sigmoid, it is able to handle more than two classes. It squeezes the output for each class betweed 0 and 1 and divides this by the sum of the outputs i.e. the probability of the input being in a particular class. The equation is given by C.45.

$$\sigma(\mathbf{z}) = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \quad forj = 1, \ldots, K. \tag{C.45}$$

The ReLU is half–rectified from the bottom as can be seen in Figure C.8. It is currently widely used within

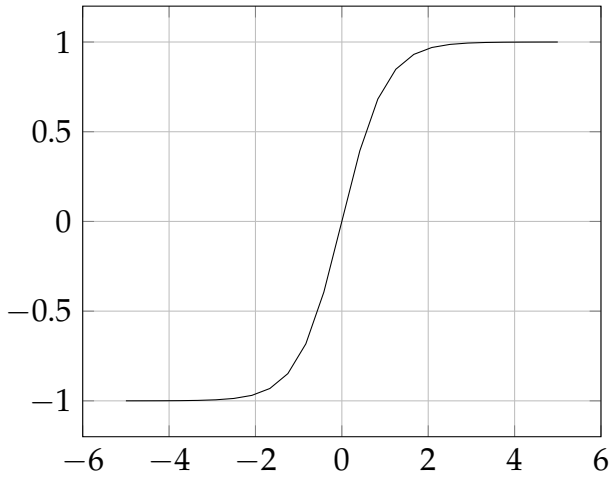


Figure C.7: Tanh Activation Function

network architectures today, and is generally used for the hidden layers of the network. It is mathematically given by a simple expression, given in Equation C.46. This means that where the input $x < 0$ the output is 0, and if $x > 0$ the output is x. ReLU causes the network to converge quicker due to its structure, where if the input is negative, the neuron would not get activated, thus making it computationally efficient. Due to this, it does not allow saturation, thus solving the vanishing gradient problem. However, in backpropagation, some gradients can move towards zero, and this may create dead neurons which will not get activated.

$$f(x) = max(0, x) \tag{C.46}$$

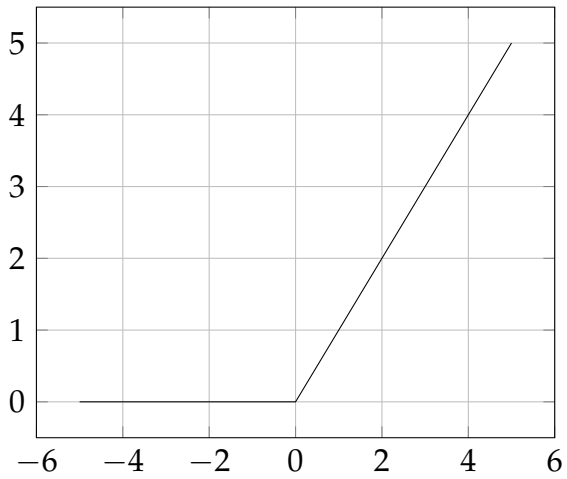


Figure C.8: ReLU Activation Function

The Leaky ReLU is a modified version of the ReLU. The original ReLU function has a gradient of $x < 0$,

creating dead neurons. Leaky ReLU (Figure C.9) addresses this problem, where values which are less than 0 have an additional linear component a, where a is a small non-zero value. Now during backpropagation, the gradient would not hit zero, thus no dead neurons.

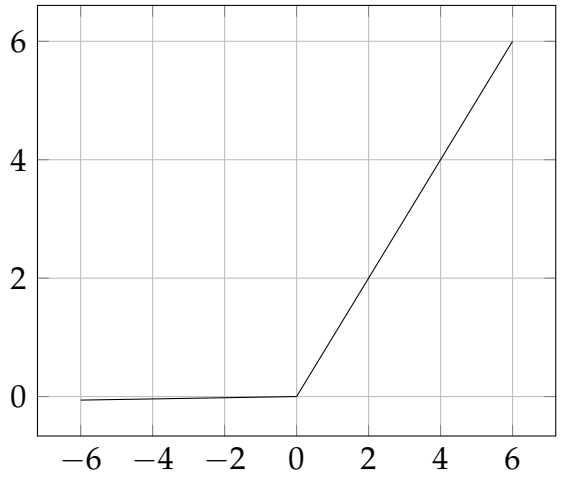


Figure C.9: Leaky Rectified Linear Unit

$$\begin{aligned} f(x) &= ax \quad x < 0 \\ &= x \quad x >= 0 \end{aligned} \tag{C.47}$$

The layers which are between the input layer and the output layer are known as the hidden layers. The number of hidden units controls the weights which are available for training, therefore controlling the complexity of the learning space of the network. An increase of hidden units combined with other regularisation techniques can increase accuracy. Neukart et al, [235], suggest to start with a hidden unit number close to the number of input neurons, and output neurons and add more if generalisation is found to be low C.48.

$$(\frac{neurons_{input}}{3} \times 2) + neurons_{output} \tag{C.48}$$

C.5 Additional Optimisation Strategies

In order to facilitate the learning process, it is usual that the initial values are normalised. However through the training process, this normalisation is lost, and the training effectively slows down as the network goes deeper.

Batch normalisation applies normalisation for every mini–batch, and back-propogates these changes through this application as well. By doing this, we have the ability to use higher learning rates without having to pay attention to the initial parameters. In addition, it also acts as a regulariser, reducing the need for Dropout, and in some cases completely taking it out of the equation.

Neelakantan et al., [236],showed that adding noise would make networks more robust to poorly initialised variables, and helps especially when training deep and complex networks. It was shown that given added noise, the model has more of a chance to escape and find a new local minima. They added noise which followed the Gaussian distribution $N(0, \sigma_t^2)$ to each of the updated gradients, shown is C.49.

$$g_{t,i} = g_{t,i} + N(0, \sigma_t^2) \tag{C.49}$$

The variance is then calculated using the following equation, C.50

$$\sigma_t^2 = \frac{\eta}{(1+t)^\gamma} \tag{C.50}$$

C.6 Hyperparameters Selection

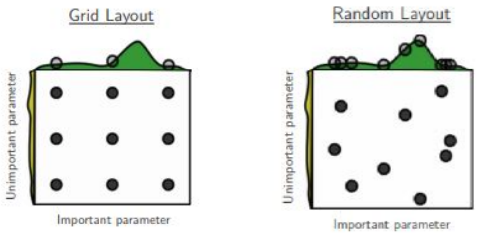
There are four methods which are used for hyperparameter selection: Manual Search, Grid Search, Random Search and Bayesian Optimisation.

Manual Search, in other words is Trial and Error, where the user would manually modify the hyperparameters, and run the algorithms. Using knowledge about the problem in question, and observation of the result, tweaks can be made to the parameters in order to find the best possible combination. However this can be prove to be time consuming, and long–winded.

Grid Search uses a simple methodology which tries all possible permutations and combinations of the hyperparameters, an example of this is shown in Fig C.10a. This is however computationally very expensive, the more hyperparameters you have, and with more dimensions the time taken to acquire the result increases exponentially.

In 2012, Bergstra and Bengio, [237], introduced Random Search where random combinations of the hyperparameters are used in order to find the best possible solution, an example is shown in Fig C.10b. For large dimensions of hyperparameters, it gives better results with less iterations. The drawback of this search it that

it yields high variance during computing, and since the selection of parameters is completely random, chance plays into finding the best combination of hyperparameters.



(a) Grid Layout *(b)* Random Layout

Figure C.10: Visual representation of types of hyperparameters search [237]

In contrast to the other forms of optimisation searches, Bayeisan optimisation keeps track of past evaluation results which are used to form a probabilistic model mapping hyperparameters to the probability of a score on the objective function: $P(score|hyperparameters)$. This model is known as a surrogate model for the objective function. There are several disadvantages to Bayesian optimisation. Results are sensitive to the surrogate model, which are generally fixed at some value. There is cost involved with searching the surrogate surface, and hence is chapter to evaluate the original problem.