ATTEMPTED MOVEMENT CLASSIFICATION OF SPINAL CORD INJURED PATIENTS COMBINING CNN AND LSTM NETWORK

KUMAR SHRESTHA

M.Sc. ENGINEERING THESIS

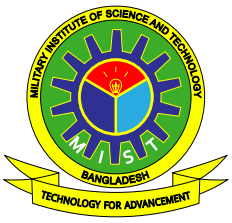
DEPARTMENT OF BIOMEDICAL ENGINEERING
MILITARY INSTITUTE OF SCIENCE AND TECHNOLOGY
DHAKA, BANGLADESH

MARCH 2023

ATTEMPTED MOVEMENT CLASSIFICATION OF SPINAL CORD INJURED PATIENTS COMBINING CNN AND LSTM NETWORK

KUMAR SHRESTHA (SN. 0421260013)

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Masters of Science in Biomedical Engineering



DEPARTMENT OF BIOMEDICAL ENGINEERING
MILITARY INSTITUTE OF SCIENCE AND TECHNOLOGY
DHAKA, BANGLADESH

MARCH 2023

ATTEMPTED MOVEMENT CLASSIFICATION OF SPINAL CORD INJURED PATIENTS COMBINING CNN AND LSTM NETWORK

M. Sc. Engineering Thesis

By

KUMAR SHRESTHA
(SN: 0421260013)

Approved as to style and content by the Board of Examination on 30 March 2023

1. Dr. Md. Asadur Rahman
   Assistant Professor
   Department of Biomedical Engineering
   Military Institute of Science and Technology (MIST)

   Supervisor
   (Chairman)

2. Col Syed Mahfuzur Rahman
   Senior Instructor and the Head of the Department
   Department of Biomedical Engineering
   Military Institute of Science and Technology (MIST)

   Member
   (Ex-Officio)

3. Lt Col Muhammad Nazrul Islam, PhD
   Instructor Class A
   Department of CSE
   Military Institute of Science and Technology (MIST)

   Member
   (Internal)

4. Dr. Mohammed Imamul Hassan Bhuiyan
   Professor
   Department of Electrical and Electronic Engineering
   Bangladesh University of Engineering and Technology (BUET)

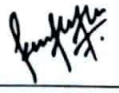
   Member
   (External)

Department of Biomedical Engineering, MIST, Dhaka

ATTEMPTED MOVEMENT CLASSIFICATION OF SPINAL CORD INJURED PATIENTS COMBINING CNN AND LSTM NETWORK

DECLARATION

I hereby declare that the study reported in this thesis entitled "Attempted Movement Classification of Spinal Cord Injured Patients Combining CNN and LSTM Network" is my original work and has been written by me. Furthermore, I have duly acknowledged all the sources of information used in this research. This research work has also not been submitted for any degree in any other university before.



Kumar Shrestha

Department of Biomedical Engineering, MIST, Dhaka

ACKNOWLEDGEMENTS

This work would not have been possible without Nepali Army. I would also like to extend my thanks to Col. Syed Mahfuzur Rahman, Head of Department, Department of BME, MIST, Dhaka, for providing numerous invaluable decisions without which this research would not have been possible. I would like to express my deep gratitude to Assistant Professor Dr. Md. Asadur Rahman, Department of BME, MIST, Dhaka, research supervisor, for his guidance and useful critiques of this research work. I am grateful to Prof. Dr. Enamul Hoque, Lt. Col. Md. Maruf Hasan, and Major Md. Ashrafuzzaman for their crucial support in every step of the journey for achieving the objectives of this research.

I would like to thank lab colleagues Ushama Shafoyat, Tanvir, Kamrul, Pte. Humaun, and all the department members for their help in administrative tasks and their support in the R& D lab. Finally, I wish to thank my friends for their support and encouragement throughout the study.

ABSTRACT

Attempted Movement Classification of Spinal Cord Injured Patients Combining CNN and LSTM Network

Electroencephalography (EEG) can be used to classify attempted hand movements of Spinal Cord Injured (SCI) patients for improving their quality of life. EEG classification with Brain-Computer Interface (BCI) allows individuals who are suffering from the most severe motor disabilities to control and direct electromechanical devices. Practical applications of BCI system require improved classification performance for attempted hand movements of SCI patients. This research aims to develop a hybrid CNN-LSTM architecture for multichannel EEG signal classification, optimize its hyperparameters, and validate performance metrics for improved classification performance. EEG data acquired from SCI patients go through filtration, downsampling, and artifact removal, followed by the generation of Time-Frequency Representation (TFR) of EEG data. Spatial encoding is done by arranging TFR data in a 2D array corresponding to spatial layout of electrodes. Spatial encoded TFR data is then fed to CNN-LSTM (Convolutional Neural Network – Long Short Term Memory) Network to obtain the final classification output. Spectral, spatial, and temporal information is vital in EEG classification. Novelty in the design of Hybrid CNN-LSTM network architecture is that it can learn to extract spectral, spatial, and temporal information and then use these learned information to improve final classification performance. Hybrid CNN-LSTM architecture achieved a classification accuracy of 92.36% using 10% of the dataset for training and 90% of the dataset for testing. This result shows 47.363% increased classification accuracy as compared to related study while also having improved generalizability. Hybrid CNN-LSTM network for EEG classification is able: to deal with artifacts in EEG data without significant loss in classification performance; to extract spectral information, spatial information, and temporal information of valuable neural impulses from EEG data. The steps of EEG classification used in this research can be used not only for attempted movements of SCI patients but also for other neurological diseases, neuroscience applications, mental workload, neuromarketing, and biometrics.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AIS	American Spinal Injury Association Impairment Scale
ALS	Amyotrophic Lateral Sclerosis
BCI	Brain-Computer Interface
CAR	Common Average Reference
CNN	Convolutional Neural Network
CVT	Complex Value Transformation
CSP	Common Spatial Pattern
DBN	Deep Belief Network
DE	Dynamic Energy
DNN	Deep Neural Network
DWT	Discrete Wavelet Transform
EEG	Electroencephalogram
EMG	Electromyogram
EOG	Electrooculogram
ERP	Event Related Potential
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
fMRI	functional Magnetic Resonance Imaging
GRU	Gated Recurrent Unit
ICA	Independent Component Analysis
IIR	Infinite Impulse Response
ITR	Information Transfer Rate
KNN	K Nearest Neighbor
LDA	Linear Discriminant Analysis
LSTM	Long Short Term Memory
MAD	Mean Absolute Difference
MEG	Magnetoencephalography
MI	Motor Imagery
MLPNN	Multi Layer Perceptron Neural Network
NLI	Neurological Level of Injury

PCA	Principal Component Analysis
PSD	Power Spectral Density
RMS	Root Mean Squared
RNN	Recurrent Neural Network
SCI	Spinal Cord Injury
SNR	Signal to Noise Ratio
STFT	Short-Time Fourier Transform
SVD	Singular Value Decomposition
SVM	Support Vector Machine
TFR	Time-Frequency Representation
TL	Transfer Learning
WT	Wavelet Transform

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

1.1 Introduction

Cervical Spinal Cord Injury (SCI) is damage to the cervical region of the spinal cord. The cervical part of the spine includes the top portion of the spinal cord which consists of seven vertebrae (C1 to C7) as indicated by Figure 1.1. Cervical SCI causes loss of motor control and/or sensation throughout the arms, trunk, and legs as brain signals cannot pass through damaged areas of the spinal cord. Motor functions along with sensory functions can be affected below the site of injury. To improve the quality of life of SCI patients, a system is required which can restore their motor functions such as hand pronation, hand supination, hand open, palmar grasp, and lateral grasp. This research deals with a classifier that can classify the neural response of attempted hand movements obtained from EEG signals of SCI patients. Such classifier can be used to provide control signals for active prostheses which can improve the quality of life of SCI patients.

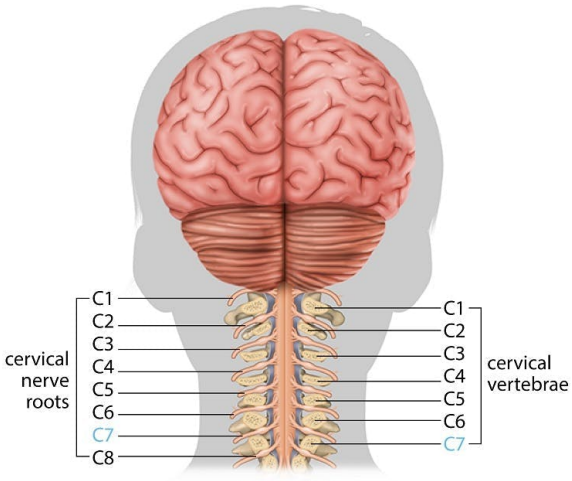


Figure 1.1: Location of cervical spinal cord and cervical vertebrae.

BCI find its application in patients who are suffering from most severe motor disabilities including SCI, amyotrophic lateral sclerosis (ALS), stroke, muscular dystrophies, chronic peripheral neuropathies, and other serious neuromuscular diseases or injuries [1, 2, 3, 4, 5]. BCI allows individuals to control electromechanical

devices by capturing neuronal impulses. These impulses are converted into instruction for controlling the output devices such as prosthetic devices, computers, and robotic systems [3].

Electroencephalography (EEG) is used to obtain neuronal impulses as it provides neurological and physiological data [6]. EEG represents the macroscopic electrical activity of the brain. This macroscopic electrical activity of the brain is recorded by using electrodes which are attached to the scalp of subjects. The recording shows up as wavy lines on an EEG recording referred to as an electroencephalogram [7]. EEG is a powerful tool in the field of clinical neurophysiology and neurology as it provides invaluable information about mental state, thoughts, imagination [8], and abnormal electrical activity of the brain [9].

A robust classifier with sufficient classification performance is required to classify EEG data. BCI generally operates in two phases: offline training phase and online operational phase. During offline training phase, the system is calibrated. During online operational phase, the system analyses brain signals and converts them into control signals [10]. After EEG signals are acquired from the user, pre-processing is done [11], features extraction is carried out [12], classification is performed [13], and application specific control signals are generated [12]. User is provided with feedback which informs them about recognition of mental command [14]. Figure 1.2 shows a general BCI system for neurorehabilitation that uses general steps discussed above.

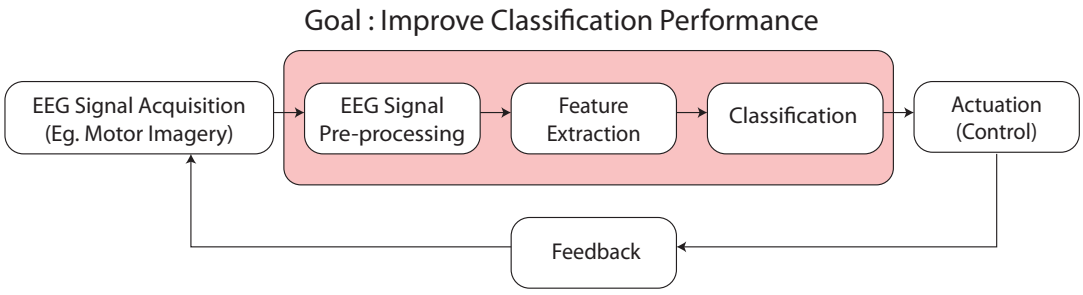


Figure 1.2: General BCI system for neurorehabilitation.

EEG classification requires an in-depth knowledge of EEG signals and their relation with neurophysiology. Clean EEG data is required to generate useful conclusions from EEG signals. Thus, limited applications of EEG classification have been seen to be available in practical scenarios. There are challenges in signal acquisition and processing, performance assessment, signal analysis, and commercialization of BCI solutions.

One of the major challenging aspects of EEG classification is that EEG signals suffer from a low Signal-to-Noise Ratio (SNR). Low SNR in EEG data is due to relatively high noise level, signals are observed through the skull, high signal contamination due to interference, and motion artifacts [13, 15]. Non-stationarity over time, within users, or between users is a major concern in EEG signals. This means EEG signals from the same user may vary among different trials [16, 17, 18, 19]. Non-stationarity of EEG signals creates difficulties in generalization because of variation of EEG signals according to the subject, time of the experiment, and state of mind [8]. This inconsistency creates complications in the generalization of BCI systems.

Artifact removal, EEG feature extraction, followed by classification of EEG signals are steps followed by EEG based BCI system. There are several considerations for signal processing for EEG based systems. Artifact Removal is complicated by low SNR. Artifacts which include motion artifacts and eye blink artifacts contribute to the noisiness of EEG. In the case of attempted hand movements, signals originate from a relatively small region of space in the head. This creates additional interference in EEG signals significantly increasing the complexity of the classification task. Furthermore, EEG signals obtained from SCI patients are more susceptible to non-stationarity and inconsistencies.

Classification accuracies for imagined movement of limbs depend vastly on: the number of output classes; the number of subjects and type of subjects; the number and placement of electrodes; and the ratio of training set to testing set. The complexity of EEG classification increases according to:

(i) Increase in number of output class

(ii) Type of subjects (e.g. SCI patients, normal human beings)

(iii) Task (Left and right hand; Hand pronation, hand supination, hand open, lateral grasp, palmar grasp)

(iv) Decrease in the ratio of training set to testing set

(v) Increase in number of electrodes

(vi) Decrease in the distance among electrodes (due to additional interference)

Study [20] has classified 5 classes of EEG data of attempted hand movements of SCI patients with maximum accuracy of 45%.

1.2 Research Motivation

Attempts have been made to classify attempted hand movements of SCI patients, but the classification performance is not adequate to be used in a real-world application. Thus, a classifier with improved classification performance is necessary for the classification of attempted hand movements (see 1.3) of SCI patients to improve their quality of life. The classifier should be able to deal with non-stationarity with less calibration time without a significant reduction in the classification performance of the system.

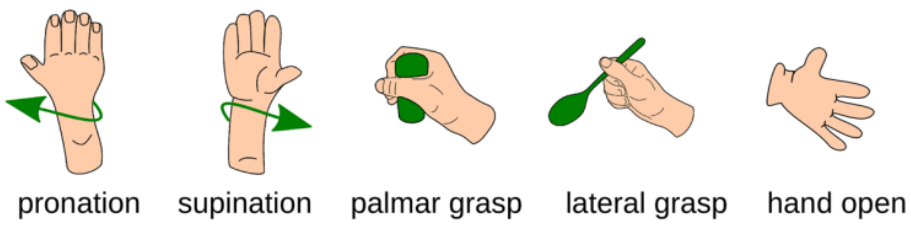


Figure 1.3: Illustration of attempted hand movements.

1.3 Research Objectives

In light of the identified issues in the area of attempted hand movement classification of SCI patients, this research aims to classify the attempted movement of SCI patients by combining Convolutional Neural Network (CNN) and Long Short Term Memory (LSTM) networks. Specific objectives of this research are outlined as follows.

(a) To develop a hybrid CNN-LSTM model for multichannel EEG signal classification.

(b) To optimize the hyper-parameters of the proposed CNN-LSTM network for EEG data of movements attempted by SCI patients.

(c) To validate and measure the classification performance of the proposed network.

The expected outcome of this work is, therefore, a new hybrid CNN-LSTM model architecture delivering improved EEG classification performance.

1.4 Contribution of this Research

This research works on achieving improved classification performance in the field of classification of EEG data from attempted movement of single hand. A novel approach is used in EEG pre-processing technique in combination with spectral, spatial and temporal feature extraction using Deep Learning for improved classification performance using fewer data for training purposes. Notable contributions in the area of EEG classification include:

(i) Effects of artifact removal techniques in EEG data

(ii) Creating strategies for extraction of spectral informations, spatial informations, and temporal informations of valuable neural impulses from EEG data

(iii) Optimization of Deep Neural Network for maximum classification performance

The concepts of the research can be used not only for attempted movements of SCI patients but also for other neurological diseases, neuroscience applications, mental workload, neuromarketing, and biometrics.

1.5 Organization of the Thesis

The remainder of this thesis is organized as follows.

Chapter 2 In this chapter, studies relating to the field of EEG classification are discussed. Topics related to this research, like EEG acquisition, EEG signal pre-processing, feature extraction, post-processing, and EEG classification are discussed with relevant literature.

Chapter 3 This chapter covers details of the steps followed for EEG classification. The chapter elaborates on strategies used in EEG acquisition, EEG pre-processing, EEG post-processing, and the use of Deep Neural Network (DNN) for maximum classification performance.

Chapter 4 This chapter discusses the evaluation criteria, performance analysis of various parameters in classification performance, CNN-LSTM network optimization, computation complexity, and final results obtained from this research. Obtained results are compared with relevant literature focusing on the improved performance of hybrid CNN-LSTM network architecture.

Chapter 5 The research work is summarized with key findings in this chapter. The limitations along with future perspectives are discussed in this chapter.

CHAPTER 2
RELATED WORKS

2.1 Introduction

Literature shows five general steps for EEG classification: EEG signal acquisition, pre-processing, feature extraction, post-processing, and classification. Various algorithms and strategies have been used in each step of motor imagery EEG classification as shown by Figure 2.1. This chapter includes: a discussion of related studies for each step of the EEG classification; a comparison of performance across various related studies; the issues in the existing literature; and the requirement for a new classifier which solves these issues.

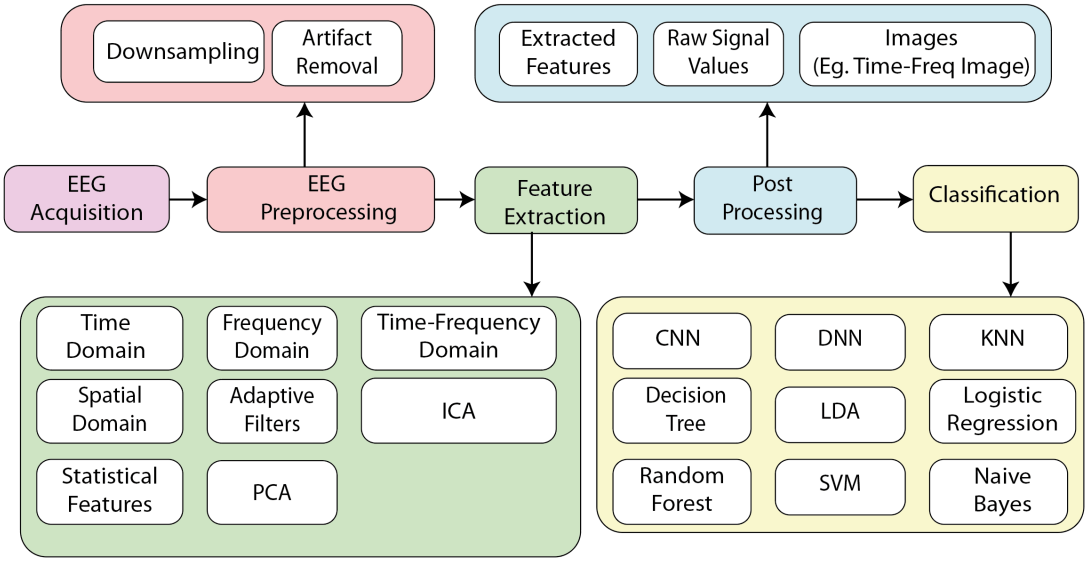


Figure 2.1: Algorithms used in different steps of EEG classification.

2.1.1 EEG Signal Acquisition

EEG signals are acquired according to the task to be classified based on concepts of neurophysiology. According to the research work [21], authors linked low-level mechanisms of EEG with more high-level explanations providing an in-depth understanding of the EEG signal. It is essential to understand the neurophysiological sources of EEG signals, propagation of generated EEG signals,

various aspects of EEG measurement, and their implication on EEG data interpretation. Usually, brain waves are measured using peak-to-peak voltage which normally ranges from 0.5 to 100 μV in amplitude. This amplitude of EEG signal is nearly 100 times lower than ECG signals. The firing of billion of neurons creates complex patterns, which can be analyzed using a mixture of various base frequencies. Researchers have classified these frequencies into frequency bands. The frequency spectrum from raw EEG signals can be used to categorise EEG signals into delta (less than 4 Hz), theta (4-8 Hz), alpha (8-13 Hz), beta (14-30 Hz), and gamma (30-80Hz) waves [22]. Alpha and beta waves are commonly used in motor imagery tasks where alpha waves correlate with eye movements and beta waves correlate with other general types of movement [23]. Heart rate, eye movement, and body temperature are used with EEG data to improve the EEG classification performance.

2.1.1.1 Motor Imagery (MI) and Attempted Movement

Neurophysiological signals generated by certain tasks must be decoded to generate control signals in BCI systems [24]. BCI systems translate brain signals into computer commands creating interaction between user and external devices. Motor Imagery (MI) signals are one of the most widely used EEG control signals. MI [25] is the process of imagining the movement of body parts without actually moving the body part. On the other hand, attempted movement is the cognitive process of attempting to move body parts while the subject may not be able to move body parts. Motor imagery is capable of creating neurological patterns related to the task in dominant sensory motor regions of the brain. In MI and attempted movement tasks, EEG signals are acquired during the imagining of a task or attempt of performing the motor movement task. Motor imagery is similar to the actual execution of the motion task. Alpha and beta frequencies of EEG signals are the most dominant in motor imagery tasks. For example, an action performed by the motor imagery task of the left-hand movement is generated from C3 region of the brain, and that of the right-

hand movement is generated from C4 region of the brain. Thus, BCI systems can regulate motor actions through imagination [26].

2.1.1.2 Electrode Placement

The number of EEG electrodes and their placement on the scalp depends on the type of tasks to be classified. For a classification task of separating 4 classes (left hand, right hand, both feet, and tongue), 22 EEG electrodes are used along with 3 Electroculogram (EOG) electrodes [27]. For subtle tasks (e.g. hand pronation) EEG signals with a high spatial resolution are required to capture required neuronal impulses. A five percent electrode placement system is used for high spatial resolution EEG acquisition. The five percent electrode placement system standardizes electrode placement in the scalp and their naming scheme [28]. Figure 2.2 shows the five percent electrode placement system which is used for high resolution EEG acquisition. 345 electrode locations are available in the five percent electrode placement system. A classification task of separating 5 classes (hand pronation, hand supination, hand open, lateral grasp, and palmar grasp) used 61 EEG electrodes with 3 EOG electrodes [20].

2.2 EEG Preprocessing

EEG signals are extremely weak signals whose amplitude ranges in microvolts. They pick up unwanted electrical and electrophysiological signals and are very susceptible to noise. There is a massive amount of interference among signals from neighbouring EEG electrodes. They suffer from artifacts such as motion artifacts, eye artifacts, muscle artifacts, and slow frequency drifts. The combined effect of all the unwanted signals results in a low SNR [29, 30, 31]. The EEG preprocessing step includes filtering, downsampling and artifact handling.

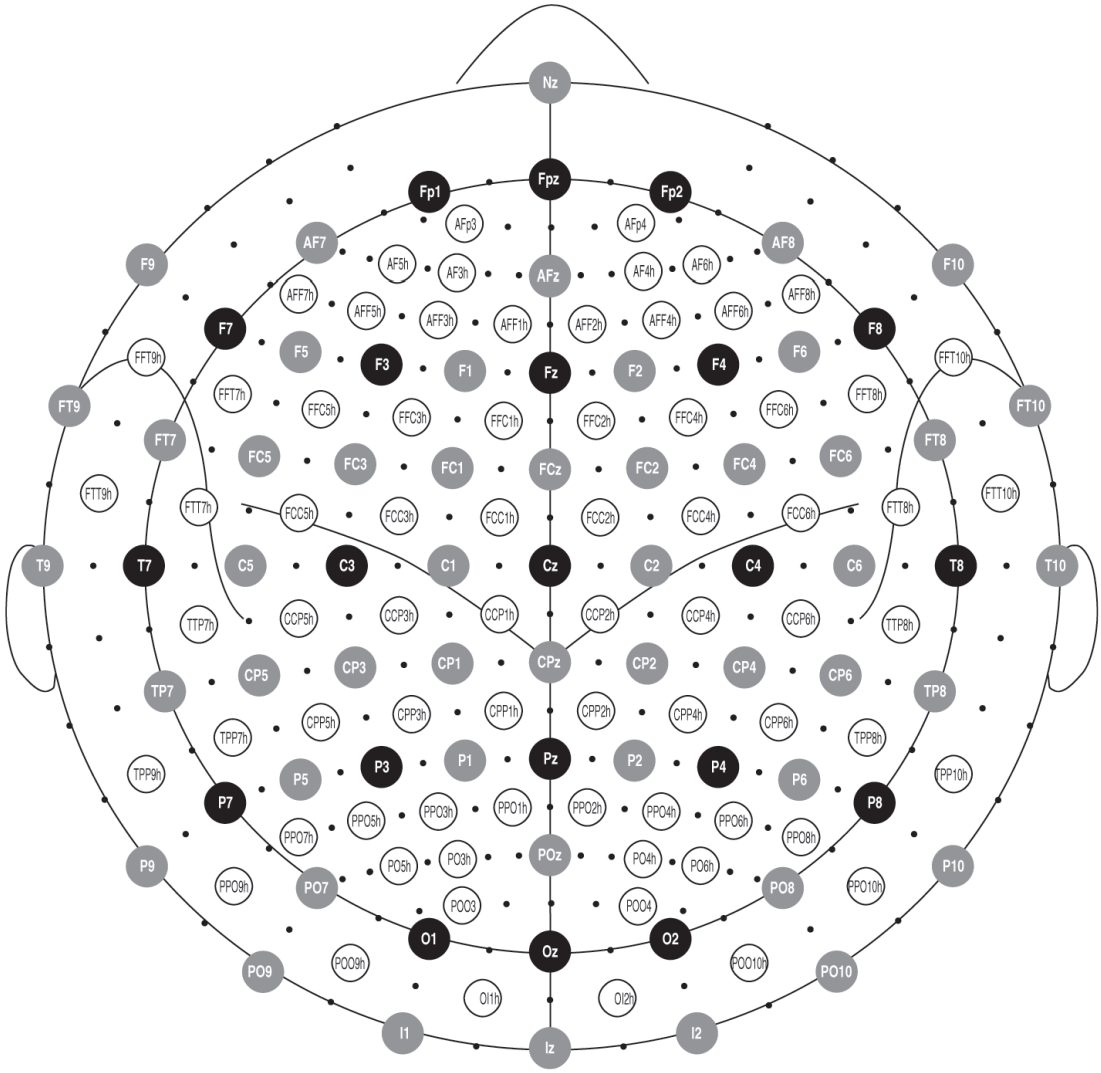


Figure 2.2: Position of electrode in five percent system used for high resolution EEG.

2.2.1 Downsampling

Sampling rate of EEG signals depends on the type of EEG acquisition system used. Sampling rate ranges according to the requirement of an application. Generally, such EEG data is resampled or downsampled at lower sampling rates. Common downsampling includes downsampling to 64 Hz [32], from 512 to 64 Hz [33], and 256 to 16 Hz [34] are proposed according to the requirement of the application.

2.2.2 Artifact Handling

Artifacts are spurious signals not related to cortical activity. They occur due to errors generated by the experimental setup, environmental noise, or biological factors. Technical/Extrinsic artifacts, and Physiological/Intrinsic artifacts are two broad categories of artifacts depending on their origin [35, 36]. Technical/extrinsic artifacts occur due to technical faults or external factors in the EEG acquisition. Technical/extrinsic artifacts include the misplacement of electrodes, powerline Interference (50–60 Hz), electromagnetic interference in the cables, etc. On the other hand, physiological/intrinsic artifacts are generated inside the human body and are not related to attempted or imagined tasks. Eye blinks (Electrooculogram (EOG) artifacts), muscle activities (Electromyogram (EMG) artifacts), cardiac activities (Electrocardiogram (ECG) artifacts), etc. are physiological/intrinsic artifacts. All these kinds of artifacts are generally handled by applying a variety of artifact removal/reduction methods such as filters, Independent Component Analysis (ICA), Principal Component Analysis (PCA), Wavelet transforms, Adaptive filters, etc. Figure 2.3 shows different kinds of artifact removal strategies used in the classification of motor imagery tasks [37]. Table 2.1 shows different artifact removal techniques used in motor imagery classification tasks [38].

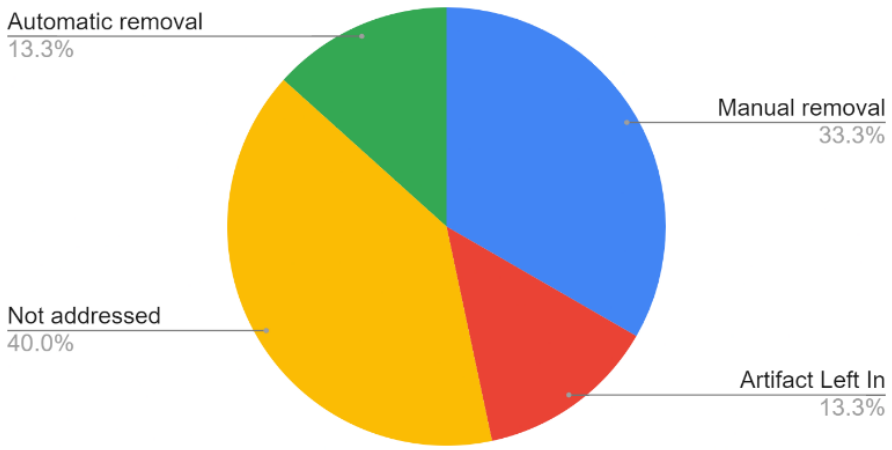


Figure 2.3: Distribution of artifact removal strategies for classification of MI tasks.

Table 2.1: Artifact removal techniques used in motor imagery classification

Artifact Removal Technique	Sources
Notch filter (50 Hz powerline interference removed)	[39, 33, 34]
Chebyshev filter 8th order, 0.01– 200 Hz	[39, 33]
Elliptic band pass filter 0.5 - 50Hz	[40]
Butterworth filter	4th order, 1– 70 Hz [39] 4th order, 0.3–100 Hz [33] 4th order, 0.3–3 Hz [33] 4th order, 0.3–3 Hz [41] 8th order, 0.01–100 Hz, 4th order, 0.2–70 Hz, 2nd order, 0.4–0.6 Hz [34]
Manual removal by visual inspection	[39, 42]
PCA	[33]
ICA	[39, 33, 34, 41]
Thresholding technique-Median absolute deviation	[33, 41]
Regression based correction method	[39, 41]

2.2.3 Feature Extraction

For effective classification of EEG signals, a certain set of features are required which represent neurophysiological activation patterns corresponding to the imagined/attempted task. These features should be distinguishable and non-redundant. Feature extraction techniques include using time-domain or frequency-domain features. Time-frequency representation is a feature extraction technique which enables spectral information to be kept along with temporal dynamics.

2.2.3.1 Time Domain Techniques and Frequency Domain Techniques

Time domain techniques include digital filters, Auto Regressive (AR) modelling, Adaptive autoregressive (AAR) modelling [43]. Frequency domain techniques have been used to extract features from MI EEG data which uses Fast Fourier Transform (FFT), Welch's method, and Local Characteristic scale Decomposition (LCD) to obtain the power spectrum. The shortcoming of spectral analysis is that it provides no temporal information. Likewise, time domain analysis doesn't use spectral features which is prone to non-stationarity problems. Studies are found which have used specific bandwidth of the EEG signal to increase SNR [37]. This method is applicable only when the frequency range of interest is known. In a review done by Craik et al. [37], approximately 50% of the studies used a low pass filter at or below 40 Hz for MI EEG classification.

Digital Filters Digital filters are used to filter frequency components from EEG signals. They can be classified into low pass filter, high pass filter, band pass filter, and band stop filter. Practically, in EEG signals, the frequency range of interest is difficult to identify. Band pass filters and notch filters are commonly used to remove artifacts, especially muscle artifacts. Notch filters are used to filter 50Hz powerline noise. FIR and IIR filters cause certain phase delays. Since phase information is very important in EEG signals phase relationship should be considered in EEG classification.

2.2.3.2 Time-Frequency Domain Techniques

Time-frequency analysis is a technique which contains both spectral and temporal information of EEG signals useful for BCI systems [14]. Study [26] showed that the utilization of correlation images of EEG channels with deep learning techniques can be used as a promising tool for dimensionality reduction of EEG data, channel fusion, and removal of artifacts. MI EEG analysis using Time-Frequency analysis include: Short Time Fourier Transform (STFT); Wavelet Transform (WT); and Discrete Wavelet

Transform (DWT). WT and DWT are very powerful techniques for feature extraction from EEG signals because different frequency bands contain valuable information about MI tasks [14, 44]. WT and DWT also give us flexibility as signals can be decomposed in multiple resolutions and multiple scale [45, 46, 47].

2.2.3.3 Spatial Domain Techniques

Spatial domain techniques for feature extraction in EEG signals include: Common Spatial Pattern (CSP), Surface Laplacian (SL), Common Average Reference (CAR), etc.

Common Spatial Pattern (CSP) CSP is a popular feature extraction method for EEG classification [48, 49, 50]. It is a spatial filtering technique which transforms EEG data into a new space by maximizing the variance of one class and minimizing the variance among different classes. It is a powerful tool as it enables the extraction of this spatial information from particular frequency bands. Generally, a frequency range of 4 Hz and 40 Hz is used which leads to the inclusion of redundant information, added noise sensitivity, and more computational time [48].

Common Average Reference (CAR) CAR is a referencing technique which eliminates the common part of the EEG signal while retaining the characteristic information related to the task for enhanced SNR [51]. This algorithm can be seen as follows:

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

Where V_i^{CAR} is the Residual clean voltage value for a specific channel,
V_i^{ER} is the original voltage value of a specific channel,
$\frac{1}{n}\sum_j^n V_j^{ER}$ is the Average voltage of all channels [52].

Surface Laplacian (SL) SL is the second order spatial derivative of the electrode potential [53]. SL aims to deal with the referencing problem. It is robust against spatial artifacts and recovers signal generated at uncovered region of the scalp by EEG electrodes.

2.2.3.4 Statistical/Wavelet Features

Different studies have used statistical features of EEG signals for their classification. Statistical features such as mean, median, variance, standard deviation, RMS, maximum, minimum, kurtosis, energy, skewness, etc. along with the combination of features have been used [38]. Discrete Wavelet Transform and Wavelet Entropy have been used to extract wavelet features for motor imagery classification [40, 54]. Behri et. al [55] have used wavelet coefficient from Wavelet Packet Decomposition as features for MI EEG classification.

2.2.3.5 Principal Component Analysis (PCA)

PCA is mainly used in the dimensionality reduction of EEG signals. It uses statistical procedures combined with techniques from linear algebra which applies an orthogonal transformation to create a series of linearly disrelated components called principal components [56]. When applied to EEG data, PCA separates EEG signals into disrelated components. These components can be analyzed and only necessary principal components can be retained. Principal components with small variance also might contain important information in EEG classification, and discarding those components might affect classification performance [57].

2.2.3.6 Independent Component Analysis (ICA)

ICA is an approach of decomposing independent sources of a complex dataset using blind source separation on the basis of statistical and linear algebra techniques [58]. It is assumed that eye artifacts, ECG, motion artifacts, and electrical noise are

generated by independent signal sources, which are non-gaussian statistically independent. These components can be separated using ICA to extract useful information related to certain tasks. ICA can be used to remove artifacts resulting in improved SNR for consecutive classification tasks [56, 58]. ICA and the Wavelet Transform (WT) used in conjunction can be used to extract spatial and time-frequency features along with various artifacts.

ICA estimates a linear transformation matrix of the input signals into source signals which are as independent as possible [59, 60, 58]. The probability distribution of independent sources is calculated using maximum likelihood solution of probability distributions of the sources.

For the likelihood, the probability density function of the sources is required, which is unknown in a blind context. Considering the source, $S_i(t)$ has density $\frac{f_i\left(\frac{*}{\sigma_i}\right)}{\sigma_i}$ with known f_i and σ_i unknown. Then $S(t)$ has density $\Pi_{i=1}^{k}\frac{f_i\left[\frac{S_t(t)}{\sigma_i}\right]}{\sigma_i}$. Since $X(t)$ is related to $S(t)$ through $X(t) = AS(t)$, the logarithm of the probability density of the data can be written as (assuming A to be invertible):

$$L_T = T\left\{\sum_{i=1}^{k}\hat{E}\ ln\left[\frac{n}{\sigma_i}f\left(\frac{e_i^T\ A^{-1}\ X}{\sigma_i}\right)\right] - ln\ |det\ A|\right\} \tag{2.2}$$

Where $\hat{E}$ = time average operator

$e_t^T = i^{th}$ column of identity matrix (of order K); transposed

The objective here is to maximize L_T to find unmixing matrix and original data. PICARD is an algorithm that converges faster than FastICA and Infomax, to obtain independent non-Gaussian sources. Details of PICARD algorithm can be found in Appendix A.

Steps in ICA A unit variance scaling is done to the data. This is followed by a pre-whitening step using PCA as follows:

(i) Data of all channels are scaled by the standard deviation.

(ii) PCA is then used to decompose pre-whitened data.

Among the resulting principal components, the smallest number of components required to explain the cumulative variance of the data is selected for ICA. Removal of unwanted independent components (artifact components) is performed and the sensor signal is reconstructed as shown in Figure 2.4 [61].

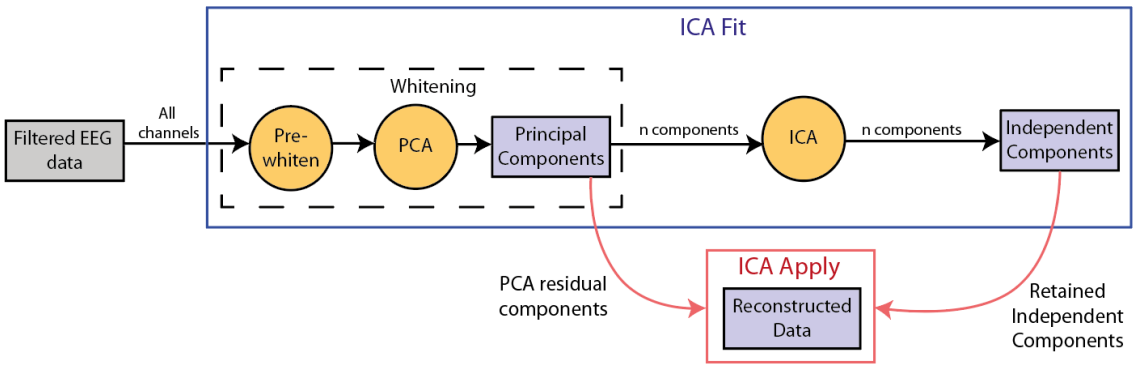


Figure 2.4: EEG data reconstruction after artifact removal using ICA.

For reconstruction of original signals excluding unwanted components:

(i) Unmix the data with unmixing matrix.

(ii) Select ICA components.

(iii) Remix the data with mixing matrix.

(iv) Restore any data which is not passed to the ICA algorithm.

Table 2.2: Comparison of various EEG preprocessing and features extraction methods

S.N.	Approach	Advantages	Disadvantages
1	ICA	• Highly efficient algorithm • Handles for large dataset very well	• More computation requirement for decomposition
2	PCA	• Reduction of dimension of features	• Not as good as ICA
3	CAR	• Best reference method	• Need sufficient head coverage
4	CSP	• Good effect on MI task • No need to select specific frequency band	• Sensitive to noise • Requires multiple electrodes
5	SL	• Robust against artifacts	• Sensitive to spline pattern • Sensitive to artifacts
6	Adaptive Filtering	• Good for signal with overlapping spectra	• Need reference signal
7	Digital Filters	• Easy removal of grounding noise	• Need signal and noise in different frequency band

S.N.	Approach	Advantages	Disadvantages
8	PCA	• Lossless dimensionality reduction	• Need more calculation for decomposition
9	PSD	• Stable features	• Not suitable for non stationary signals • Unable to analyze time domain signals
10	AR	• Need small sequence of data • Decrease spectra loss issue • Good frequency resolution	• Not suitable for stationary signals • Presents low performance once estimated model mistakenly elected
11	FFT	• Suitable for stationary signals • Suitable for narrowband signals	• Not suitable for non stationary signals • Great noise sensitivity
12	WT	• Works well with non stationary signals • Signal analysis in time and frequency domain	• Deficient particular mode to apply to prevalent noise

2.2.4 Post-Processing

As the features are extracted, they are organized and further necessary processing is done to extract information from these features for the classification of EEG. This step is referred to as post-processing. Post-processing includes feeding the classifier with calculated features, signal values, or images constructed from extracted features. Distribution of usage of different types of input formulation for MI classification tasks is given in Figure 2.5.

Figure 2.5: Input formulations across MI EEG classification studies. The outer circle shows specific input formulation and inner circle shows general input formulation techniques.

2.3 EEG Classification

Classification is a process in which one can group and classify different cases based on learning and development by using the available data and knowledge experience. Classifiers are used to recognize the EEG patterns generated by users based on extracted EEG features. Figure 2.6 shows that Linear Discriminant Analysis (LDA) and Support Vector Machine (SVM) classifiers are mostly used in MI EEG classification tasks. Other classifiers include, Extreme learning machines, Logistic regression, Decision Tree, Linear discriminant analysis, Neural networks, Convolutional Neural Network (CNN), Deep Neural Network (DNN), Long short-term memory, Naïve Bayes, Support vector machine, and Random forest have been used [37]. Table 2.3 shows a comparison of the advantages and disadvantages of different classification algorithms for motor imagery EEG classification [62].

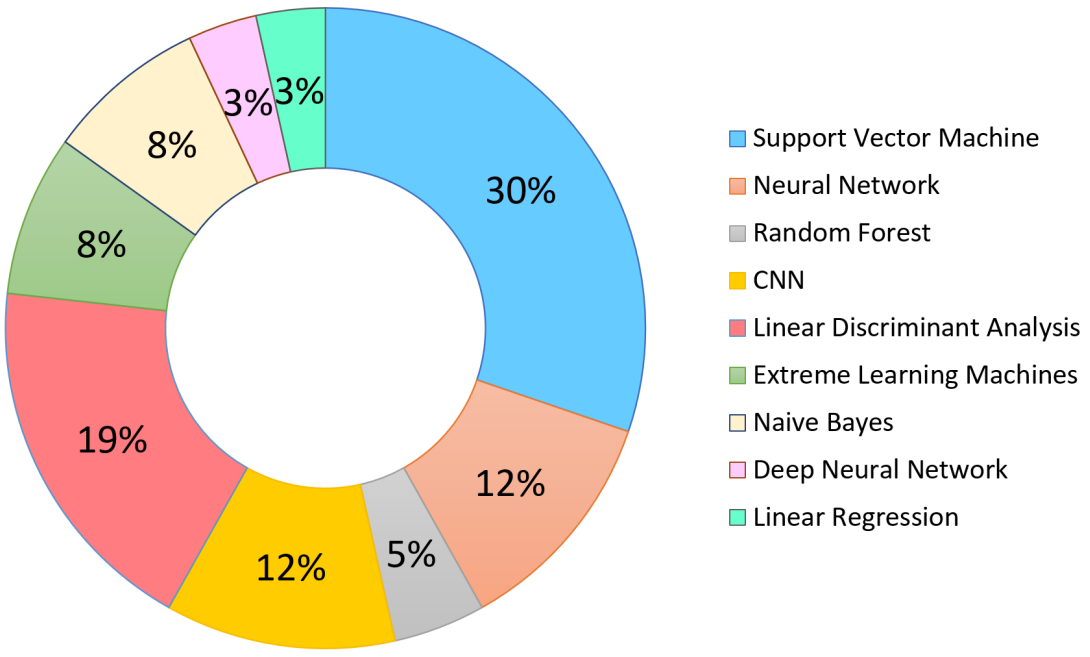


Figure 2.6: Various algorithms used in MI EEG classification tasks.

Table 2.3: Comparison of different EEG classification algorithms

S.N.	Approach	Advantage	Disadvantage
1	ANN and DL	• High accuracy • Flexible structure	• Performance depends on architecture
2	K-NN	• Easy to understand • Simple to implement	• Sensitive to redundant and irrelevant features • Requires large storage
3	LDA	• Easy to use • Low computational requirement	• Requires a linear system
4	SVM	• Performance is better than other linear classifiers	• Low computational complexity
5	NB	• Easy to understand	• Independent variables

Recent advancements in EEG classification have seen deployment in deep learning architectures due to the availability of big EEG datasets. This led to improvement in the performance of classification for Motor Imagery EEG signal and their analysis. Deep learning has been used for the classification of various motor imagery tasks [63, 64, 65].

2.3.1 Neural Networks and Deep Learning

Wang et al. [66] found that the CNN architecture significantly outperformed the pure LSTM-RNN architecture. Thus, CNN or DBN is recommended for motor imagery task classification. On the other hand, deep learning approach learns directly from data. Multiple layers of trainable feature extractor modules are arranged in certain layered architecture in DNN. Generally, learnt features are related to increasing levels of concepts. The concept of DNN enables the interpretation of big datasets. Hybrid

architectures of neural networks have outperformed architecture with a single type of neural network. In Hybrid architectures, 90% of hybrid architectures combined convolutional layers with another type of layer (eight out of those nine using one or more RNN layers [37]. Although task specific hybrid designs are not common, results show LSTM RNN modules outperformed standard and GRU-RNN modules with consistent improvements.

2.3.2 Performance Evaluation of Related Motor Imagery Classifications

For the valid and accurate evaluation of various related studies done in the field of MI/attempted movement classification, multiple aspects of the classification task should be considered. These considerations include:

- Number of output classes
- Type of tasks to classify
- Number of subjects and their type (Normal or real-world patients)
- Training and testing set used
- Accuracy achieved

Table 2.4 shows details on output classes, tasks, subjects, training set, testing set, and accuracy achieved on various classification studies for motor imagery task [37, 24].

Table 2.4: Classification accuracy of various motor imagery classification studies

Output Classes	**Strategy**	**Tasks**	**Subjects**	**Train Set**	**Test Set**	**Accuracy (%)**	**Study**
2	Sparse group Representation	Left hand, Right hand	9	0.80	0.20	78.2	[67]
2	Capsule network	Left hand, Right hand	9	0.57	0.43	78.44	[68]
2	CNN	Left hand, Right hand	9	0.90	0.10	78.93	[69]
2	LDA, SVM	Left hand, Right hand	9	0.60	0.40	79.33	[70]
2	Sparse Bayesian Learning	Left hand, Right hand	9	0.90	0.10	81.7	[71]
2	SVM with PSO	Right hand, Foot	5	0.60	0.40	91.6	[72]
2	CNN	Left hand, Right hand	14	0.75	0.25	92.73	[42]
2	RLDA, LDA	Left hand, Right hand	7	0.90	0.10	93.04	[73]
2	SVM	Right hand, Foot	5	0.90	0.10	96.02	[74]
3	SLDA	Attempted Hand Movement – Hand open, Palmar grasp, Lateral grasp	10	0.90	0.10	53	[20]

Output Classes	**Strategy**	**Tasks**	**Subjects**	**Train Set**	**Test Set**	**Accuracy (%)**	**Study**
3	Weighted Naïve Bayes	Left hand, Right hand, Foot	5	0.50	0.50	86.38	[75]
3	Naïve Bayes	Left hand, Right hand, Foot	5	0.90	0.10	96.36	[76]
3	CNN-LSTM	Left hand, Right hand with eye movement, None	4	0.90	0.10	96.5	[77]
3	CSP + LSTM network	Left hand, Right hand, Foot	52	0.90	0.10	96.91	[78]
4	Recurrent CNN	Left hand, Right hand, Feet, Tongue	9	0.90	0.10	45	[79]
4	CNN	Left hand, Right hand, Feet, Tongue	9	0.80	0.20	61	[80]
4	Spiking Neural Network	Left hand, Right hand, Feet, Tongue	7	0.10	0.90	63.6	[81]
4	CNN	Left hand, Right hand, Feet, Tongue	9	0.50	0.50	70.6	[82]

Output Classes	**Strategy**	**Tasks**	**Subjects**	**Train Set**	**Test Set**	**Accuracy (%)**	**Study**
4	Deep RNN with Spatial frequency features	Left hand, Right hand, Feet, Tongue	9	0.50	0.50	73.56	[83]
4	CNN	Left hand, Right hand, Feet, Tongue	9	0.90	0.10	74.46	[84]
4	Multi layer CNN fusion	Left hand, Right hand, Feet, Tongue	20	0.84	0.16	75.7	[85]
4	CNN-LSTM	Left hand, Right hand, Feet, Tongue	9	0.80	0.20	76.62	[86]
4	RNN-GRU	Left hand, Right hand, Feet, Tongue	9	0.60	0.40	77	[83]
4	SVM	Left hand, Right hand, Feet, Tongue	3	0.80	0.20	77.96	[87]
4	LDA	Left hand, Right hand, Feet, Tongue	9	0.88	0.12	78.82	[88]

Output Classes	**Strategy**	**Tasks**	**Subjects**	**Train Set**	**Test Set**	**Accuracy (%)**	**Study**
4	DBN	Left hand, Right hand, Feet, Tongue	9	0.90	0.10	80.4	[89]
4	SVM	Left hand, Right hand, Feet, Tongue	9	0.90	0.10	83.82	[90]
4	Extra-Trees algorithm	Left hand, Right hand, Feet, Tongue	9	0.90	0.10	84	[91]
4	Ensemble k-nearest neighbor (k-NN)	Left hand, Right hand, Feet, Tongue	9	0.90	0.10	90.32	[92]
4	CNN-LSTM with Transfer learning	Left hand, Right hand, Feet, Tongue	9	0.80	0.20	92	[93]
4	RLDA	Left hand, Right hand, Feet, Tongue	7	0.90	0.10	93.04	[73]
4	CNN-LSTM	Left hand, Right hand, Feet, Tongue	9	0.90	0.10	95.62	[94]

Output Classes	**Strategy**	**Tasks**	**Subjects**	**Train Set**	**Test Set**	**Accuracy (%)**	**Study**
4	CNN	Left hand forward, Left hand backward, Right hand forward, Right hand backward	1	0.36	0.64	100	[95]
5	SLDA	Attempted Hand Movement : Pronation, Supination, Hand open, Palmar grasp, Lateral grasp	10	0.90	0.10	45	[20]
5	RNN-LSTM	Eye closed, Both fist, Both feet, Left fist, Right fist	12	0.86	0.14	68	[96]

2.4 Chapter Summary

According to recent related literature in the areas of classification of motor imagery tasks, approximately 50% of the studies are seen to have classified 4 classes of motor imagery tasks. Classification accuracy for four class classifications has reached a maximum of 95.62% [94] for motor imagery tasks (left hand, right hand, tongue, and feet) across multiple subjects. Only two studies are seen to have classified five motor imagery or attempted movement tasks. For five class classifications, the maximum classification is reduced to 68% [96]. Only two studies have classified attempted movement of subtle movements (hand pronation, hand supination, hand open, lateral grasp, and palmar grasp) [20, 97] as compared to motor imagery task of bigger movements (left hand, right hand, tongue movements, and feet). For the classification of attempted hand movements (hand pronation, hand supination, hand open, lateral grasp, and palmar grasp) the classification accuracies are as follows:

- Maximum classification for two classes is 70.09% [97].
- Maximum classification for three classes is 53% [20].
- Maximum classification for five classes is 45% [20].

Drastic reduction is seen in maximum classification accuracy for subtle hand movements as compared to whole limb movement. This reduction can be attributed to the additional complexity in the classification of attempted movement tasks invited by: interference among EEG electrodes; higher spatial resolution and higher number of EEG electrodes; and real-world SCI patients. When it comes to real-world SCI patients, there is a significant increase in non-stationarity problems as compared to normal human beings. Thus, it is seen that a robust classifier is required that can classify subtle hand movements of real-world SCI patients for the improvement of the quality of life of SCI patients. Additionally, this classifier should have:

(i) Good classification performance across non-stationarity of EEG data

(ii) Have low calibration time (using less training data as compared to testing data).

(iii) Can extraction of frequency information, localization information, and time information of valuable neural impulses from EEG data

CHAPTER 3
MATERIALS AND METHODS

3.1 Introduction

This research aims to classify77he attempted movement of SCI patients by combining CNN and LSTM networks. General steps of EEG classification are: EEG acquisition, EEG preprocessing, Feature extraction, Post-processing, and Classification, as shown by Figure 2.1. To overcome the issues seen in current literature as seen in Section 2.4, following steps of EEG classification are performed:

(i) **EEG Acquisition**: Dataset is derived from research done by Ofner et al. [20]. It uses a 64 electrode high resolution EEG acquisition system according to five percent electrode placement system for EEG [28] is used.

(ii) **EEG Preprocessing**: Filtering, downsampling, and artifact removal using ICA is performed.

(iii) **Post-Processing**: A time-frequency representation of cleaned EEG signal is generated which is sent to spatial encoding step. After spatial encoding, it is forwarded to datagenerator for feeding input data and label data to CNN-LSTM network.

(iv) **Feature Extraction and Classification**: Feature extraction and Classification is performed by a DNN. The DNN is composed of CNN and LSTM networks. Initial part of DNN is composed of convolutional layers which extract spectral features, temporal features, and spatial features. Following feature extraction layers, LSTM layers are used to learn temporal dynamics of extracted features. Finally, outputs from LSTM networks are fed to end classifier made up of convolutional layers and dense layer to classify information obtained from LSTM into five different classes.

The approach of using ICA for artifact removal and Time-Frequency Representation of EEG data is used for reducing loss of classification performance due to non-stationarity

in EEG signals. Furthermore, the novelty of this research compared to related studies lies in the use of 2D spatial encoding according to the physical layout of the EEG sensors along with 3D Convolutional layer and 2D LSTM layer has led to efficient learning of spectral, temporal, and spatial dynamics of EEG data. Likewise, spectral features, spatial features, and temporal features are extracted and classified by DNN itself. Figure 3.1 shows the proposed steps followed for the classification of attempted movement of SCI patients. This chapter further elaborates, each step of the proposed EEG classification in detail.

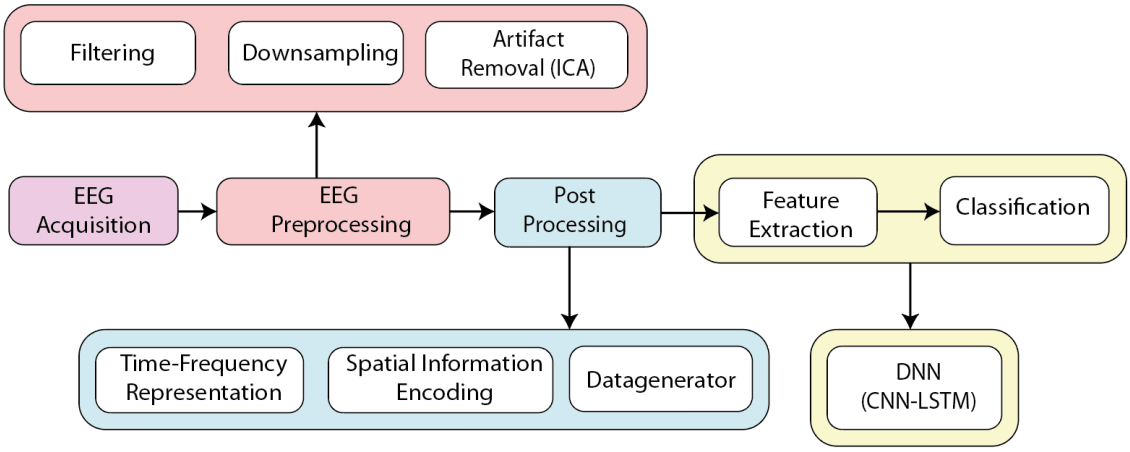


Figure 3.1: Block diagram of proposed steps of EEG classification.

Maximum classification accuracy is achieved by developing a hybrid CNN-LSTM network architecture and optimization of its hyperparameters. EEG pre-processing and post-processing are done with MNE-Python [61] while feature extraction and classification is done using TensorFlow [98]. Figure 3.2 depicts the system flow diagram of the proposed CNN-LSTM classifier for EEG classification.

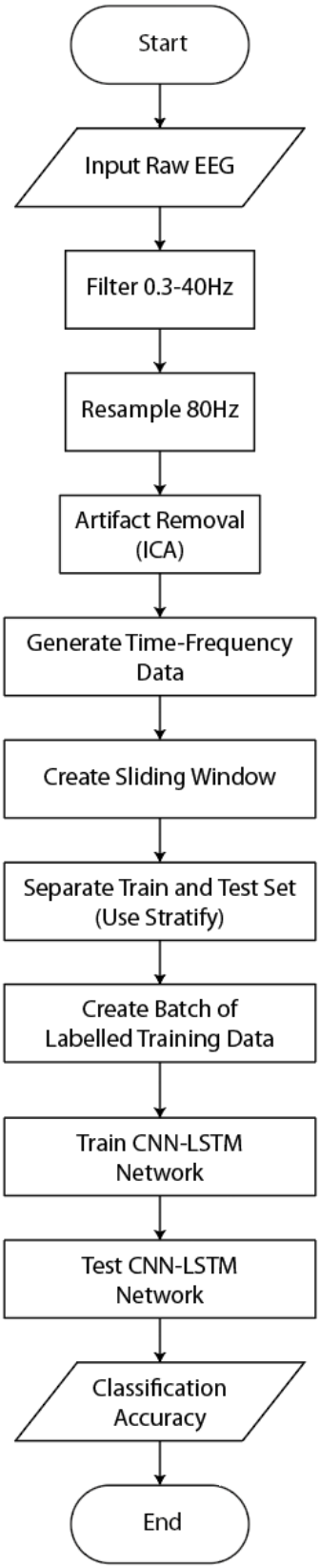


Figure 3.2: System flow diagram of hybrid CNN-LSTM network for EEG classification.

3.2 EEG Acquisition

The dataset of attempted hand movement of Spinal Cord Injured patients is collected from the work done by Ofner et al. [20]. The dataset consists of EEG signals from 10 participants with cervical spinal cord injury (SCI). Patients are aged from 20 to 69 years. Details on age, dominant hand, time since lesion, sex, tested hand, American Spinal Injury Association Impairment Scale (AIS), and Neurological Level of Injury (NLI) is shown in Table 3.1. Five types of hand movement attempt (pronation, supination, hand open, palmar grasp, and lateral grasp) were recorded as shown in Figure 3.3. These five types of movements relate to 5 different classes for the classification of EEG signals. The type of hand movement is referred to as a cue for a trial.

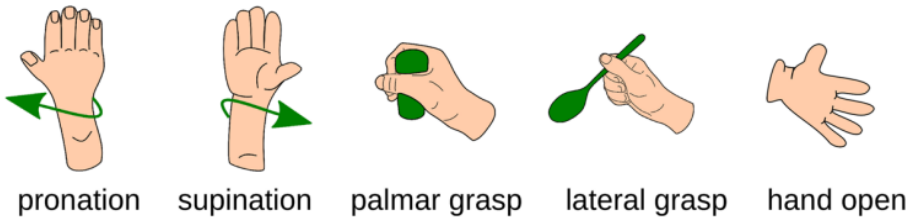


Figure 3.3: Illustration of attempted hand movements.

Table 3.1: Details of SCI patients

Participants	Sex	Age	Dominant Hand	Tested hand	Time Since Lesion (Months)	AIS	NLI
P01	M	23	Right	Right	11	B	Sub C6
P02	M	42	Right	Right	10	D	Sub C1
P03	M	62	Right	Right	7	B	Sub C5
P04	F	20	Right	Right	201	B	C5
P05	M	57	Right	Right	9	A	Sub C4
P06	M	78	Right	Right	7	D	Sub C5

Participants	Sex	Age	Dominant Hand	Tested hand	Time Since Lesion (Months)	AIS	NLI
P07	M	27	Right	Left	4	C	Sub C4
P08	M	69	Right	Right	24	B	C7
P09	M	53	Right	Right	74	A	Sub C4
P10	M	55	Right	Right	23	A	Sub C6

3.2.1 Experimental Setup

Each of the SCI patients was put in front of a computer screen with their arm resting on a pillow or their stage, or on a table. The SCI patients carried out the instructions provided on the computer screen. When the trial started, a fixation cross and a beeping sound were presented. SCI patients were asked to concentrate their attention on the cross. Cross was displayed during the whole trial period of five seconds to avoid eye movements as shown in Figure 3.4. At-2 second, a cross was presented together with a beeping sound; the class cue was shown at 0 second; and the SCI patients executed/attempted the movement. After the trial, a small break with an arbitrary time duration of 1s to 3s followed. Likewise, SCI patients were instructed to avoid swallowing and blinking during the trial period. The class cue was displayed from 0 to 3 seconds time frame(i.e. until the end of the trial).

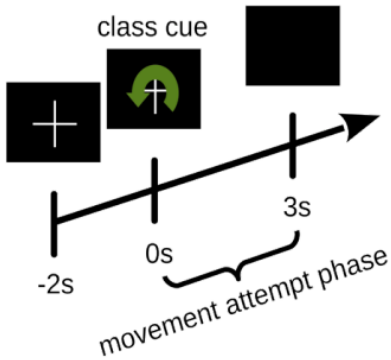


Figure 3.4: Trial sequence.

According to the residual motor capacities of subjects, subjects were informed to execute the corresponding movement right after the class cue appeared. Also, they were informed to avoid unrelated movements. Subjects went back to their original beginning position after the trial period. A break with an arbitrary period of one second to three seconds followed after each trial. There is a total of 9 runs with 40 trials per run, i.e. 72 trials per class for each case. Table 3.2 show the type of run according to the run number for all cases.

Table 3.2: Description of different types of runs of EEG acquisition

Run Number	**Run Type**
1	Eye movements
2	Rest
3	Attempted movement
4	Attempted movement
5	Attempted movement
6	Attempted movement
7	Attempted movement
8	Eye movement
9	Rest
10	Attempted movement
11	Attempted movement
12	Attempted movement
13	Attempted movement
14	Eye movement
15	Rest

3.2.2 Electrode Placement

EEG data is acquired using 61 EEG electrodes covering frontal, temporal, central, and parietal areas using five percent electrode placement system. Additionally, an

electrooculogram (EOG) is measured using 3 electrodes which are placed above the nasion and below the outer canthi of the eyes. Signals were recorded using a g.GAMMAsys/g.LADYbird active electrode system (g.tec medical engineering GmbH, Austria) and a four 16-channel g.USBamps biosignal amplifiers. Signals were sampled at 256 Hz and a band-pass filter was applied from 0.01 Hz to 100 Hz (8th order Chebyshev filter). A notch filter at 50 Hz was applied to suppress power line interference. Reference was placed on the left earlobe and ground was placed on AFF2h. Table 3.3 shows the channel number and corresponding channel labels used in the experimental setup. Obtained EEG data are stored in GDF (General Data Format for Biosignals) format. Figure 3.5 shows the topographic plot of the location of electrodes in the patient's scalp.

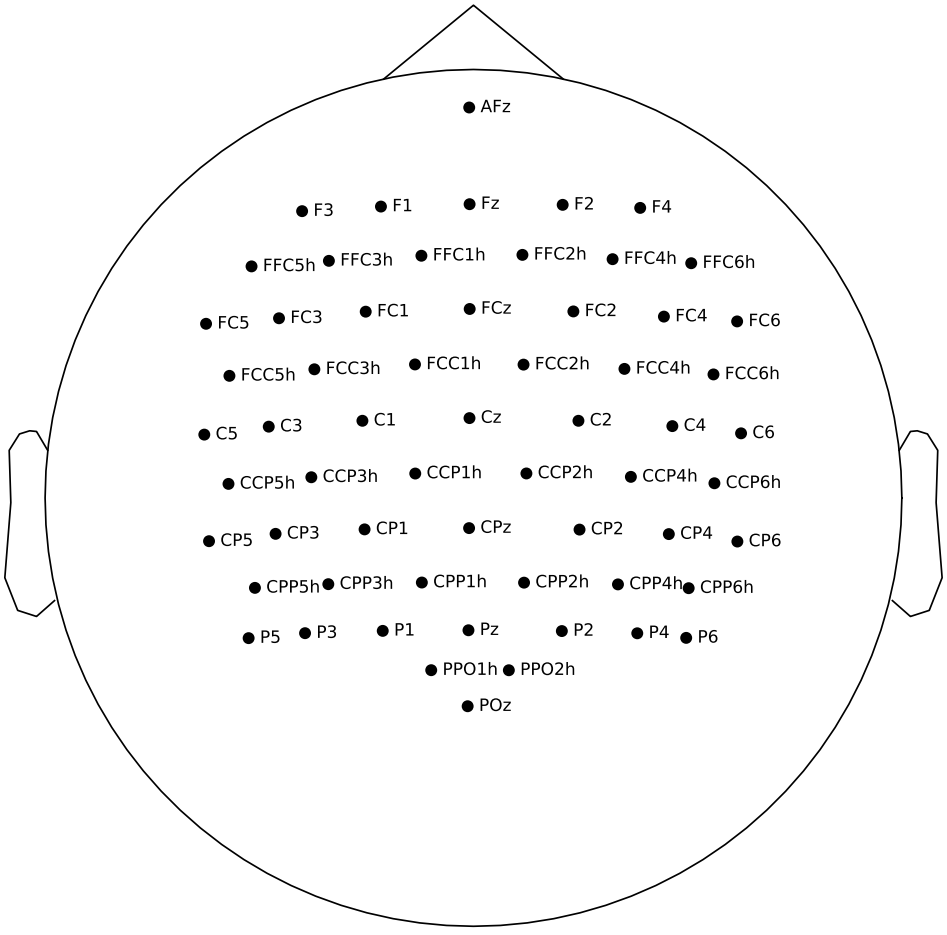


Figure 3.5: Topographic plot of electrode location.

Table 3.3: Channel number and corresponding channel labels

Channel	Label	Channel	Label	Channel	Label	Channel	Label
1	AFz	17	FC2	33	CCP5h	49	CPP2h
2	F3	18	FC4	34	CCP3h	50	CPP4h
3	F1	19	FC6	35	CCP1h	51	CPP6h
4	Fz	20	FCC5h	36	CCP2h	52	P5
5	F2	21	FCC3h	37	CCP4h	53	P3
6	F4	22	FCC1h	38	CCP6h	54	P1
7	FFC5h	23	FCC2h	39	CP5	55	Pz
8	FFC3h	24	FCC4h	40	CP3	56	P2
9	FFC1h	25	FCC6h	41	CP1	57	P4
10	FFC2h	26	C5	42	CPz	58	P6
11	FFC4h	27	C3	43	CP2	59	PPO1h
12	FFC6h	28	C1	44	CP4	60	PPO2h
13	FC5	29	Cz	45	CP6	61	PPOz
14	FC3	30	C2	46	CPP5h	62	EOG left
15	FC1	31	C4	47	CPP3h	63	EOG mid
16	FCz	32	C6	48	CPP1h	64	EOG right

3.3 EEG Pre-processing

EEG pre-processing step consists of Filtering, artifact removal using ICA, and Downsampling. This step is done to clean EEG signals and get rid of unnecessary artifacts that pose a problem in the classification process.

3.3.1 Filtering

A filter is used to remove or attenuate unwanted frequencies of a signal to increase the SNR of the EEG signal. Effects of filtering are cautiously considered to minimize

filter artifacts. Distortion of EEG signals which causes artifacts are reduced. For this purpose, one pass zero phase non-causal Finite Impulse Response (FIR) filter is used. One pass filters are used so as to reduce computational complexity while zero phase non-causal filters are used to reduce the risk of distortion of the EEG data. Frequency dependent delays are introduced by a non linear phase, which causes distortion in signals even for spectral components within the passband of the filter. Thus, a single pass zero phase bandpass FIR filter is used, since phase information is critical to EEG classification. This makes FIR filter non-causal, meaning output not only depends on past inputs but also future inputs. Hamming window is used for desired pass band ripple and stop band attenuation with minimum distortions and filter artifacts created by the filter. A 0.3Hz low pass band edge and a 40 Hz high pass band edge with -6dB transition cutoff bandwidth is used.

3.3.1.1 Finite Impulse Response (FIR) filters

The transfer function of digital filter in frequency domain is given by:

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + ... + b_M z^{-M}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + ... + a_N z^{-N}} \tag{3.1}$$

$$H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}} \tag{3.2}$$

whereas in time domain, the numerator coefficients b_k along with denominator coefficients a_k can be used to obtain filtered output data y(n) in terms of the input data x(n) given as:

$$y(n) = b_0 x(n) + ... + b_m x(n - M) - a_1 y(n - 1) - ... - a_n y(n - N) \tag{3.3}$$

$$y(n) = \sum_{k=0}^{M} b_k x(n - k) - \sum_{k=0}^{N} b_k x(n - k) \tag{3.4}$$

The output of the filter at a particular time n is determined by the sum over:

- the numerator coefficients b_k multiplied by previous input values $x(n-k)$.
- the denominator coefficients a_k multiplied by previous output values $y(n-k)$.

FIR filters use a finite number of numerator coefficients b_k (where $a_k = 0 \forall k$), and thus each output value y(n) depends only on the M previous input values.

Windowed sinc FIR filters are based on the sinc-function in time domain approximating a rectangular magnitude response in frequency domain. For finite filter orders, impulse response has been windowed by a window function for reduction of passband and stopband ripple (equal for windowed sinc filters). Filtering is implemented as convolution of the filter input with the impulse response. The transition bandwidth, pass band ripple, and stop band ripple are used as a function of filter order.
A one pass, zero phase, non causal bandpass FIR filter is used whose filter characteristics are listed below:

(i) Windowed time domain design method

(ii) Hamming window with a 0.0194 passband ripple and 53 dB stopband attenuation

(iii) Low passband edge: 0.30 Hz

(iv) Low transition bandwidth: 0.30 Hz (-6 dB cutoff frequency: 0.15 Hz)

(v) Upper passband edge: 40.00 Hz

(vi) Upper transition bandwidth: 10.00 Hz (-6 dB cutoff frequency: 45.00 Hz)

(vii) Filter length: 2817 samples

Figure 3.6 and 3.7 shows frequency response, Figure 3.8 shows impulse response, and Figure 3.9 shows step response of FIR filter.

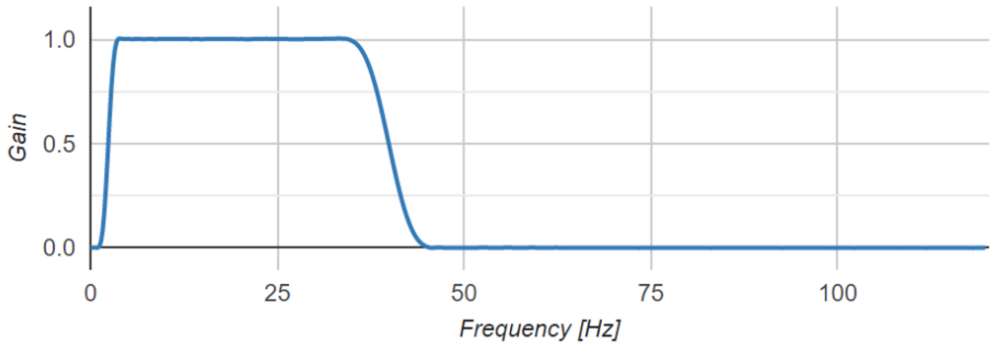


Figure 3.6: Frequency response of FIR filter.

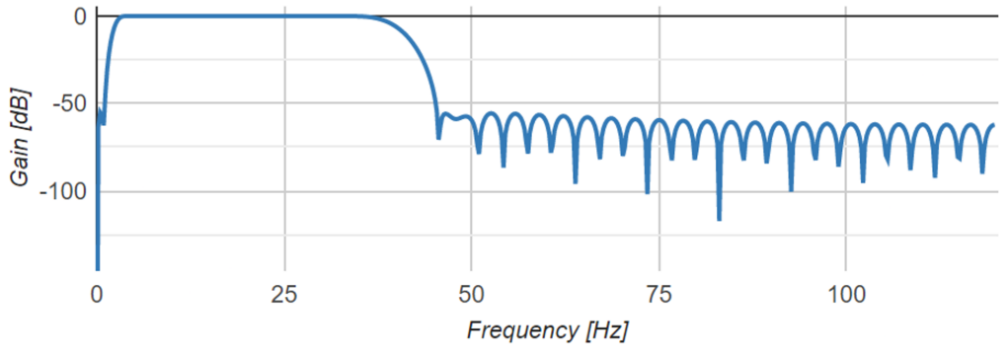


Figure 3.7: Frequency response (dB) of FIR filter.

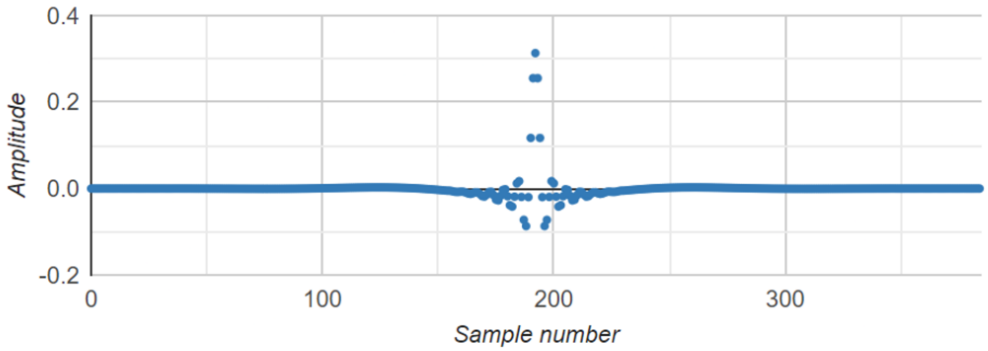


Figure 3.8: Impulse response of FIR filter.

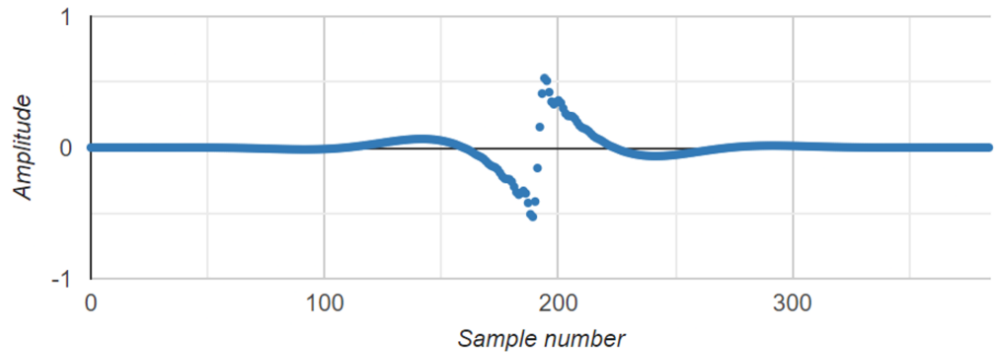


Figure 3.9: Step response of FIR filter.

3.3.2 Downsampling

EEG signal is downsampled to 80Hz sampling rate from 256 Hz original sampling rate. 80 Hz is selected considering upper passband edge of 40 Hz as Nyquist frequency.

3.3.3 Artifact Removal using Independent Component Analysis (ICA)

There is a crucial requirement of an effective method for removing artifacts and unwanted signals from EEG data of real-world SCI patients. ICA models data as a linear mixture of non-gaussian independent sources. It is used to separate statistically independent and non-gaussian components from the contaminated observed signal. Each independent component separated by ICA is analyzed. Unwanted components are rejected manually. Clean EEG signals are reconstructed using the remaining components. The process of independent component separation and reconstruction of a clean signal is depicted in Figure 2.4. ICA provides the ability to extract and then remove artifacts and noise from EEG signals. Figure 3.10 shows independent components of an EEG acquisition run separated by ICA with rejected components grayed out. Each component is analysed manually and marked for rejection. To achieve desired results, following configuration of ICA is used:

(i) PCA Components = 0.99 (Retaining at least 99% of variance in data)

(ii) Iteration = 1000

(iii) Method = Picard

PICARD is used because it converges faster than Infomax and FastICA, and is more robust than other algorithms when the sources are not completely independent similar to real EEG data [59]. PCA components explaining 99% variance of the data are used because missing valuable signal components is not desirable in cases like EEG where SNR is very low.

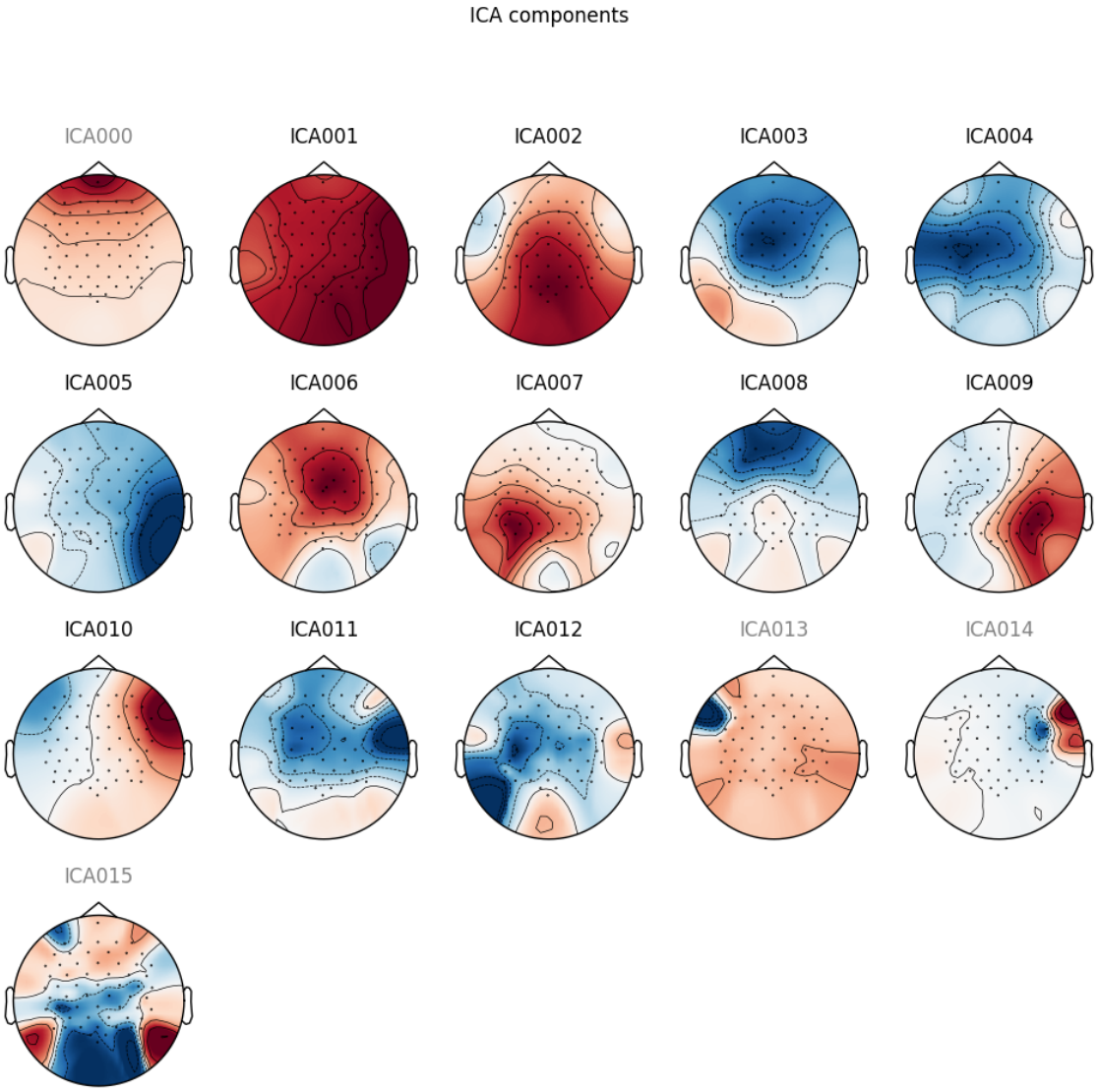


Figure 3.10: ICA components of a run (Bad components grayed out).

Neural response related to the experiment are considered a good component while the rest are considered bad components of EEG data which are identified manually. To identify good or bad components, various aspects of the components need to be considered simultaneously. These aspects depicted by subplots (refer Figure 3.11 for subplot) include: spatial information (described by topoplot); time series information (described by Epochs image and ERP); frequency information (described by frequency spectrum); and finally variance of the component with respect to Epochs.

3.3.3.1 Good Components of ICA

Good ICA components can be identified by analyzing subplots according to:

(i) **Spatial contour** of the component (Clear depiction of origination of the component from a single concentrated point as seen in top left topoplot of Figure 3.11).

(ii) Clear indication of **Event related potential** seen in Epochs image and time series graph (Epoch Image and time graph on top right corner of Figure 3.11).

(iii) Indication of strong **Alpha frequencies** (around 10Hz) seen in Frequency spectrum (Bottom left graph of Figure 3.11)

(iv) Indication of variation of the component in different epochs (Bottom right scatter plot of Figure 3.11). 8 different epochs (single cue experiment) show greater than unit variance giving a clear indication that this change occurred due to a particular cue. It is to be noted that there are 8 different epochs for each cue in an experimental run, and the ICA is applied to separate each experimental run.

Figure 3.11 shows ICA component related to neural response of attempted movement from EEG data.

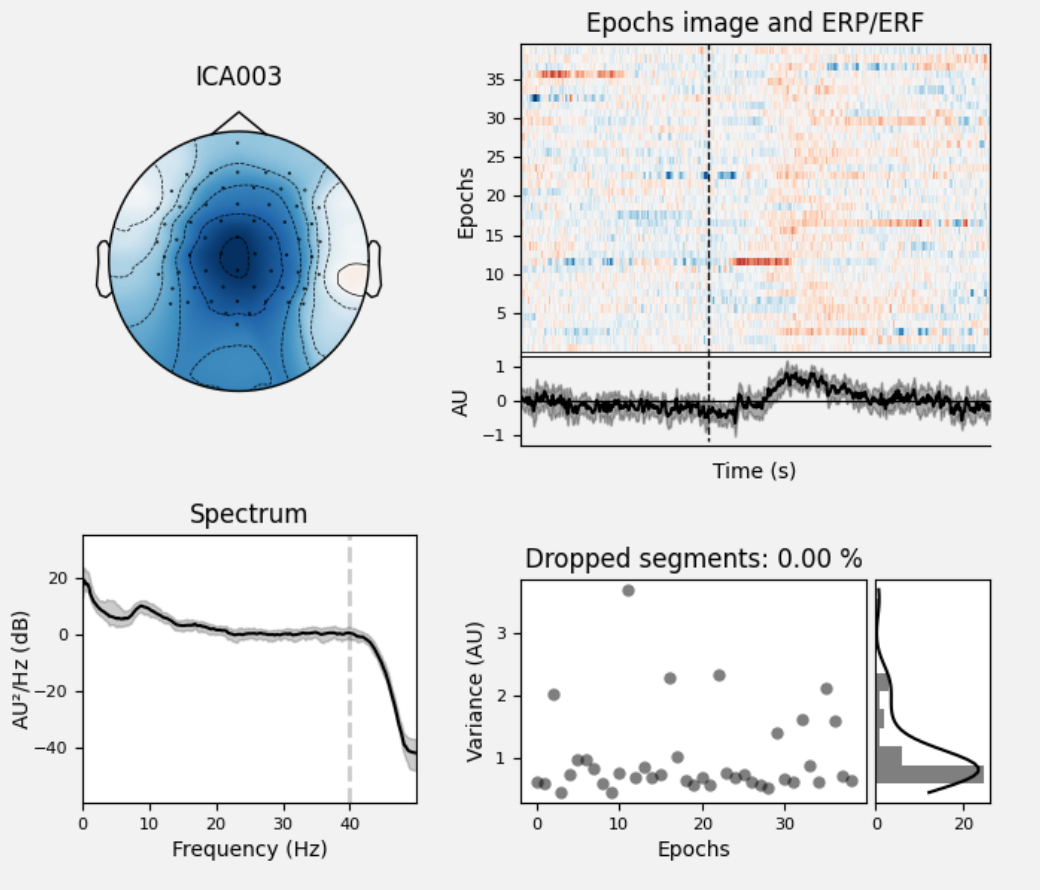


Figure 3.11: ICA component related to neural response of attempted movement.

3.3.3.2 Unwanted Components of ICA

Unwanted components of ICA include:

- Eye blink
- Motion artifacts
- Unwanted signals according to the region of origin
- Noise
- Slow frequency drift
- Muscle artifacts
- Signals which occur before the cue is shown to the patient

- Components whose variance distribution does not comply with the distribution of experimental cues. If there is only one epoch having more than a unit variance of the components, it is highly unlikely to be related to 8 epochs of the same cue. For 8 epochs of the same cue, ideally, there should be 8 epochs having more than unit variance as depicted by Figure 3.11.

These unwanted components can be separated by analyzing subplots (topoplot, epoch image, time graph, frequency spectrum, and variance scatter plot) of the component. Each unwanted component has its own characteristic feature that can be distinguished using these subplots. Often all the subplots need to be considered to select unwanted components.

Eye Blink Eye blinks can be distinguished by their spatial contour (originating from nasion toward inion) and rapid change in component amplitude in Epochs image as shown by Figure 3.12.

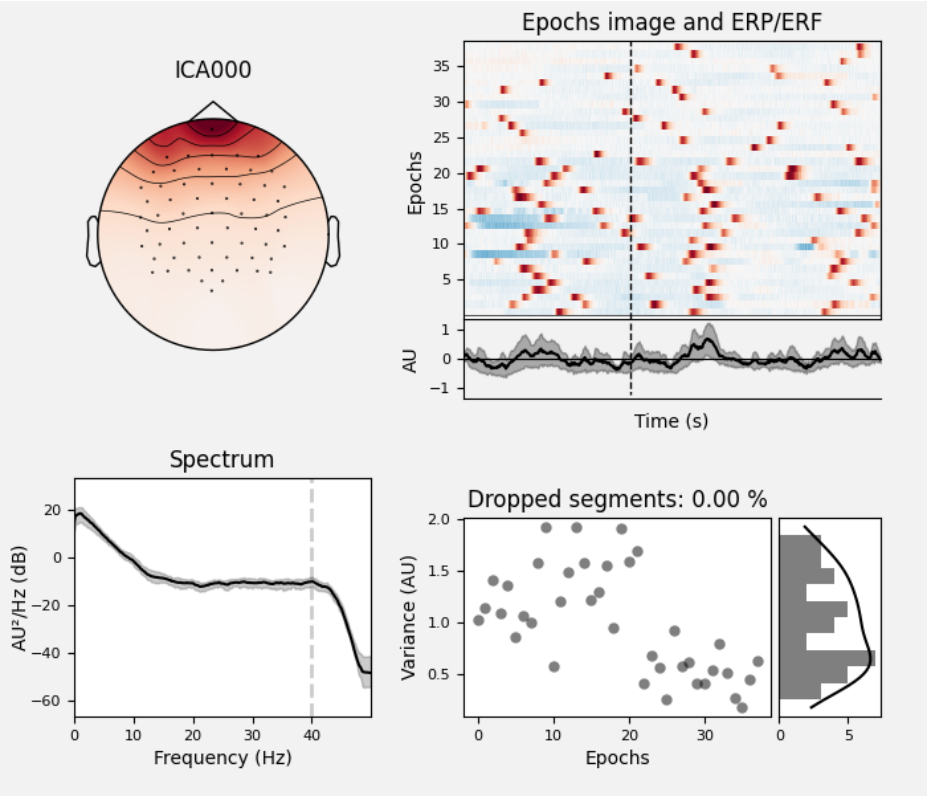


Figure 3.12: Eye blink component.

Motion Artifacts Motion artifacts due to electrode wobbling are identified by rapid changes in amplitude that are not part of cue signals. They can be identified from Epochs image as shown in Figure 3.13. Other types of motion artifacts include patient movement. Patient movement can be confirmed by a spike in variance in a specific region of time which is not related to cue signals. Figure 3.14 shows a spike in variance from 10 to 20 epochs showing patient movement.

Unwanted Signals According to Region of Origin Some components are generated at the very edge of the topoplot and also have some characteristic occurrences in variance scatterplot. These components have a very high variance for a single epoch thus making these components undesirable. Figure 3.15 shows such a component.

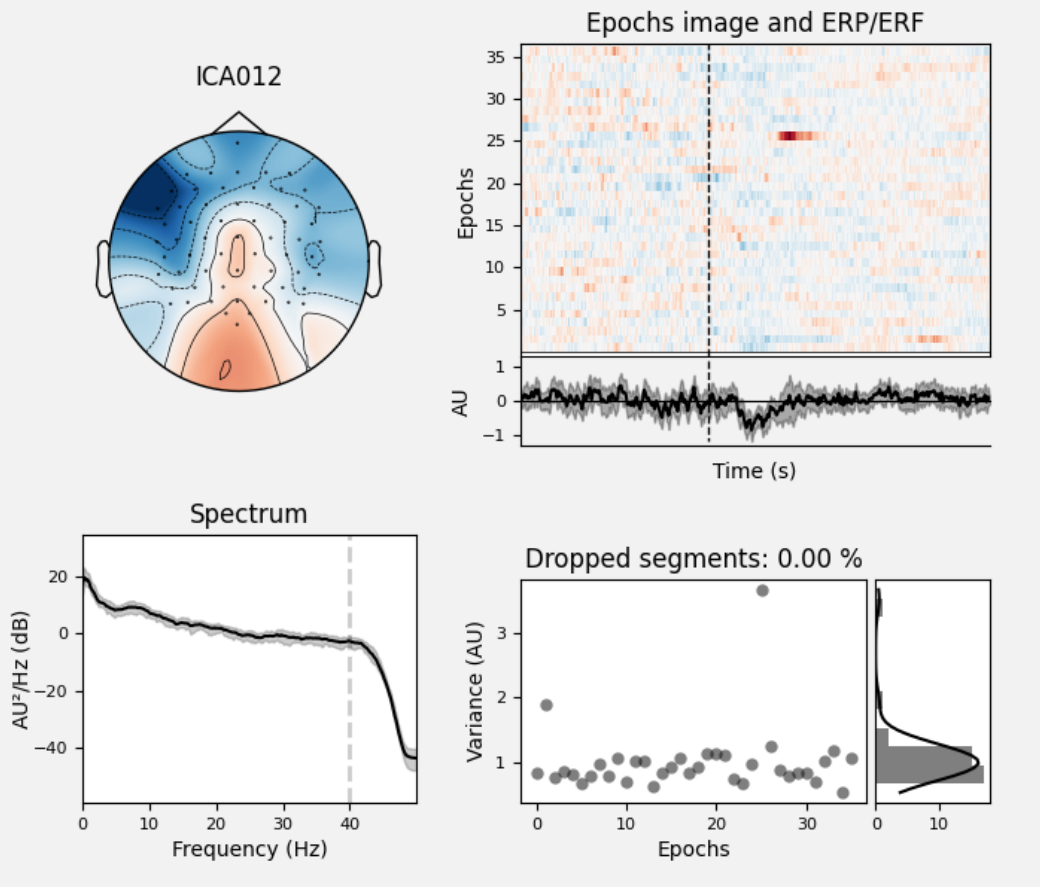


Figure 3.13: Motion artifact component separated by ICA.

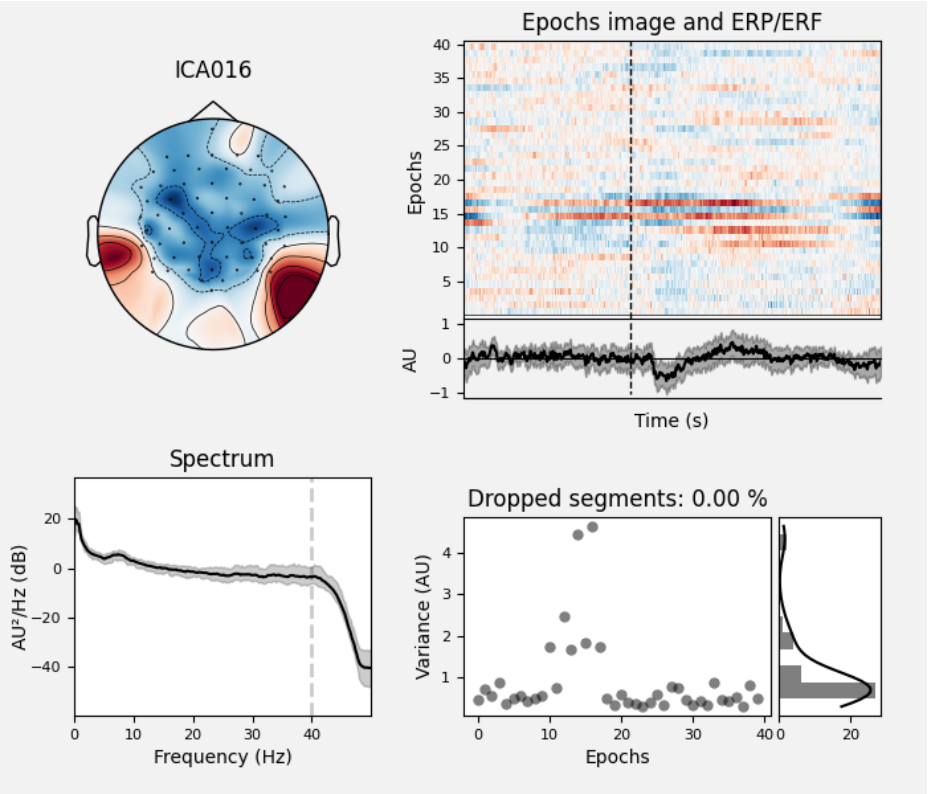


Figure 3.14: Patient movement component separated by ICA.

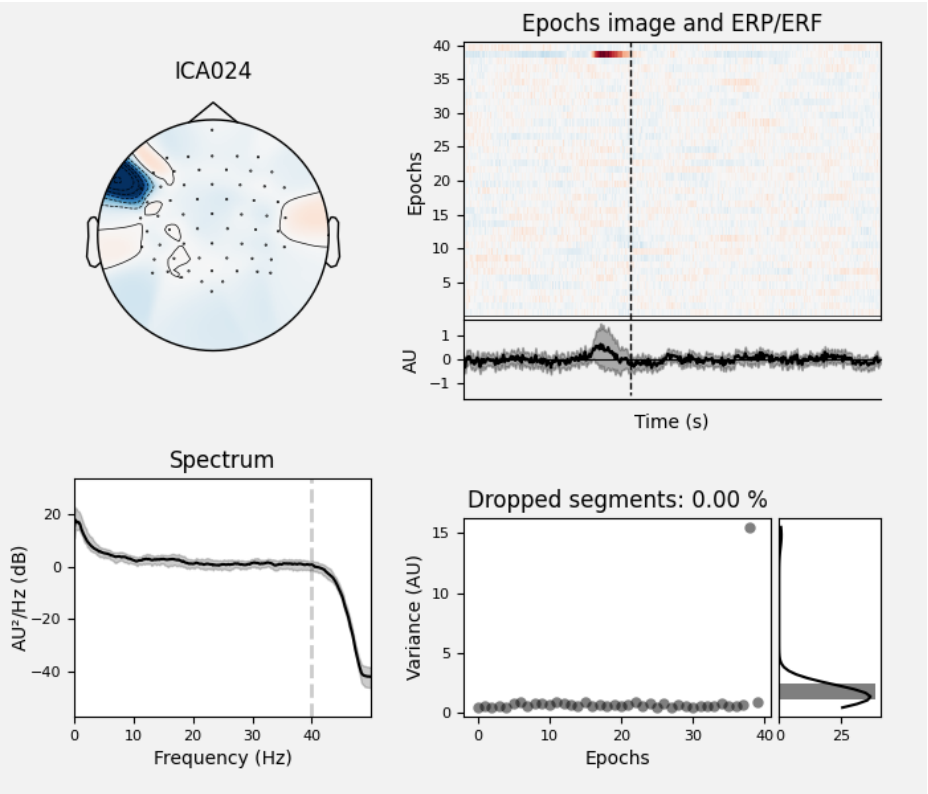


Figure 3.15: Unwanted component according to the region of origin and frequency of occurrence.

Noise and Slow Frequency Drift Noise can be identified by the lack of event related potential response and their random nature seen in the amplitude of the component seen in the time graph (refer Figure 3.16). Slow frequency drifts can be identified from a linear relationship seen in the variance of the component and Epoch number as seen in Figure 3.16. Also, the time series plot of the component and Epoch image can be used to find noise components and muscle artifacts.

Muscle Artifact Muscle artifacts can be distinguished by their frequency spectrum (occurrence of signals from 7 Hz to 150 Hz) [99]. Also, their variance scatter plot is no substantial relation with a specific cue, and there is a lack of event related potential in the time graph. Muscle artifacts can also originate from any muscle in head as seen in topoplot which include origination of signal from edges. Figure 3.17 shows muscle artifacts.

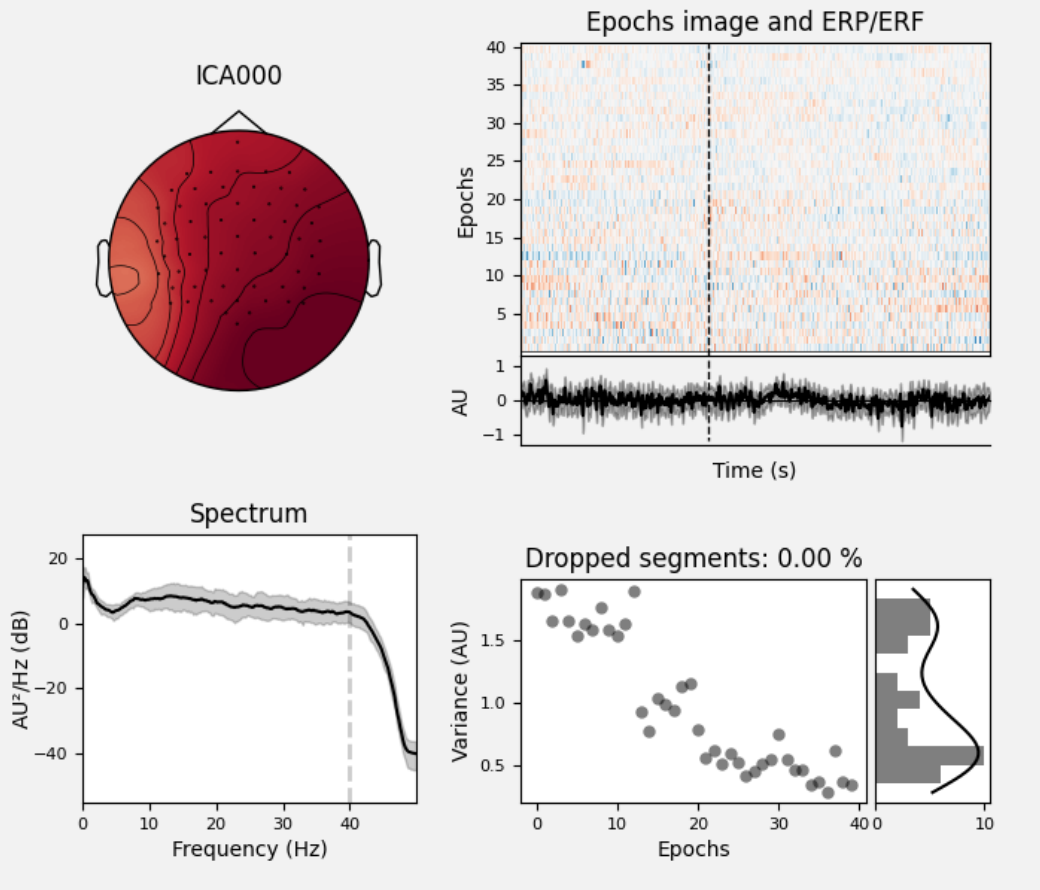


Figure 3.16: Unwanted component according to noise and slow frequency drift.

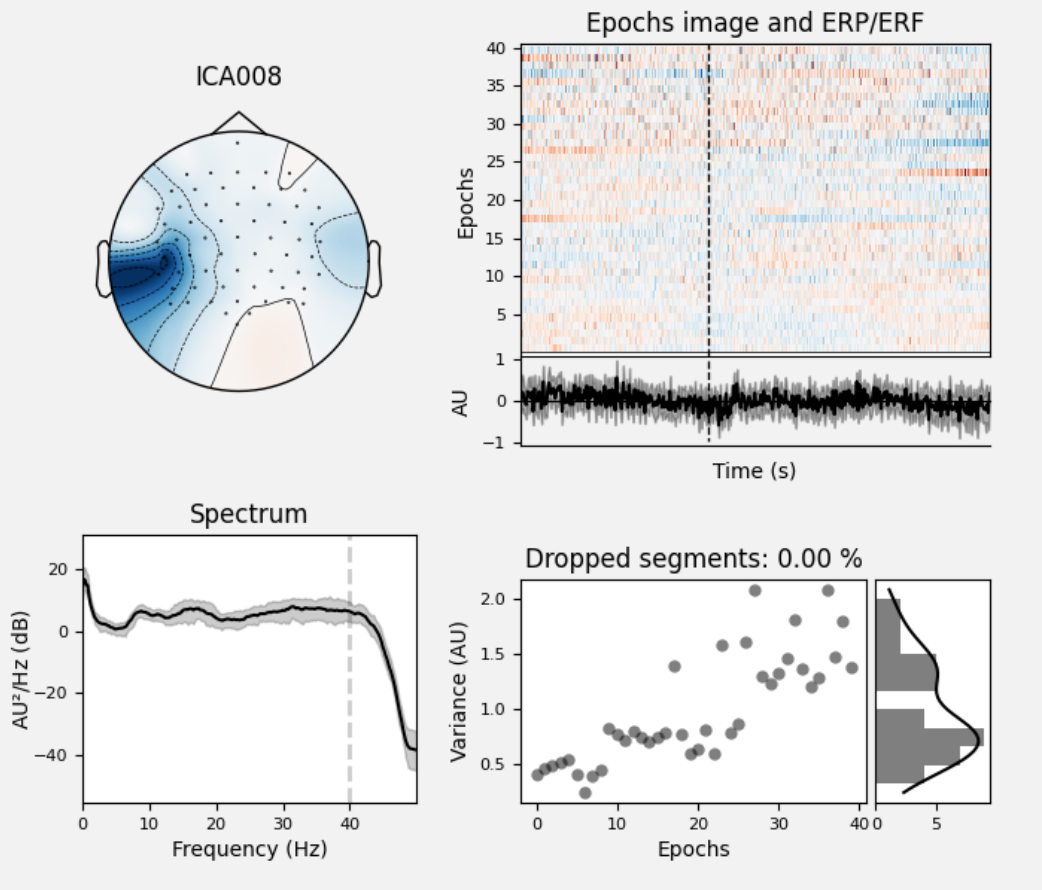


Figure 3.17: Muscle artifact component separated by ICA.

3.4 Post-Processing

Post-processing is the step of manipulating the pre-processed EEG data so that it can be utilized for improved classification performance. This step is depicted by Figure 3.18 which includes:

(i) Generation of **Time-Frequency Representation** (TFR) of cleaned EEG signal.

(ii) Conversion of TFR data to data for hybrid CNN-LSTM classifier. This step includes the following sub-steps:

(a) **Data loader**: Arrangement of all EEG channels into two dimensional array to encode spatial information of the channel.

(b) **Sliding window**: Generation of the sliding window to minimize the effect

of non-stationarity of EEG signals.

(c) **Stratified train-test split**: Splitting dataset into training and testing set. Splitting is done in such a way that training and testing data contain equally distributed data of multiple cues.

(d) **Data generator**: It randomly selects training (containing input data and corresponding labels) and testing data (containing input data) from separated training and testing set and feeds it into hybrid CNN-LSTM network.

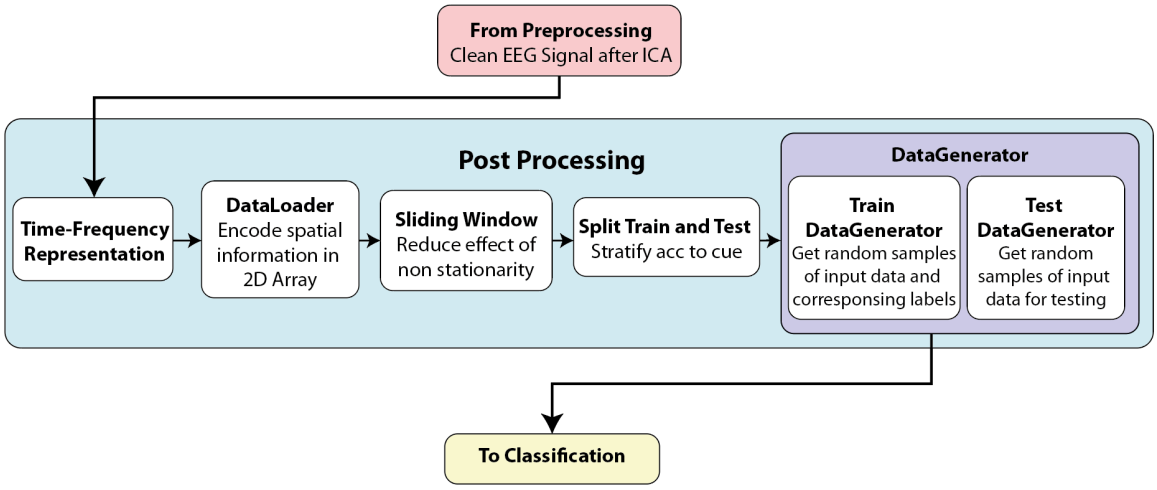


Figure 3.18: Post-processing step of EEG classification.

3.4.1 Time-Frequency Representation

TFR provides a time-varying energy of the signal in each frequency band. The EEG signal is convoluted by complex Morlet's wavelets $w(t, f_0)$ [100] having a Gaussian shape both in the time domain (having Standard Deviation σ_t) and in the frequency domain (Standard Deviation σ_f) around its central frequency $f_0 : \omega(t, f_0) = A\ exp\left(\frac{-t^2}{2\sigma_t^2}\right)\ exp\left(2i\pi f_0 t\right)$ with $\sigma_t = \frac{1}{2\pi\sigma_f}$. Figure 3.19 shows Morlet wavelet with its sine and cosine components. Wavelets are normalized so that their total energy sums to 1, the normalization factor A being equal to $(\sigma_t\sqrt{\pi})^{\frac{-1}{2}}$.

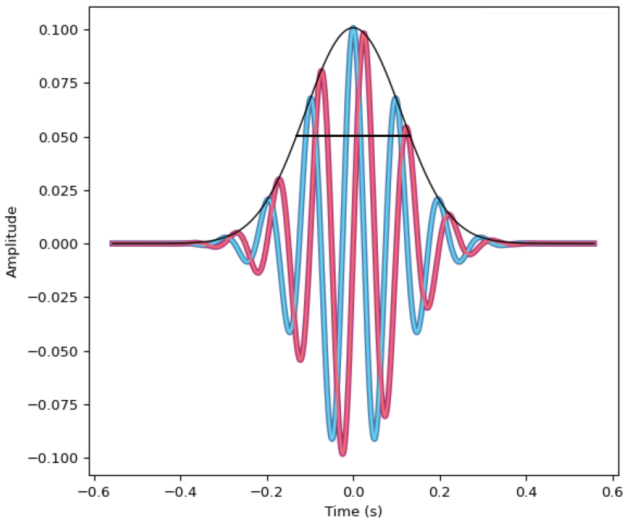


Figure 3.19: Morlet wavelet.

Here the time-varying energy $E(t, f_0)$ of the signal in a frequency domain is the square normalization of the result of the convolution of a complex wavelet $\omega(t, f_0)$ with the signal $s(t)$:

$$E(t, f_0) = \|\omega(t, f_0) \times s(t)\|^2 \tag{3.5}$$

TFR of the signal is obtained by the convolution of the signal by a family of wavelets. In morlet wavelet analysis, the oscillation that is defined by 'n_cycles' is tapered by a Gaussian taper where the edges of the wavelet are attenuated according to the Gaussian curve. The number of cycles used for gaussian taper is not necessarily helpful for understanding the amount of temporal smoothing that has been applied [101]. In spectrotemporal analysis, the temporal and spectral resolution are interrelated. Longer temporal windows allow more precise frequency estimates. On the other hand, shorter temporal windows smear frequency values while providing more precise timing information. In that case, the full width at half maximum (FWHM) of the wavelet can be useful. The FWHM of the wavelet at a specific frequency is defined as:

$$FWHM = \frac{n\_cycles \times \sqrt{2\,ln\,2}}{\pi \times freq} \tag{3.6}$$

For a fixed FWHM value, n_cycles is defined as a function of frequency which implies temporal window decreases in length with an increase in frequency. For simplicity n_cycle = $\frac{freq}{2}$ is used with 40 frequency signals to generate 40 different sliding temporal windows. Time-frequency representations are computed by convoluting sliding temporal windows with original signals.

TFR image for each channel for each trial as shown by Figure 3.20. In the case of this research, there are 72 trials for each cue for each patient. Each trial contains a 5 second time series data of 61 EEG channels simultaneously. Thus, for each trial, there are 61 TFR images. The hybrid CNN-LSTM classifier should extract features and classify them into 5 different cues by training from these 64 images of a single trial. Settings used for Time-Frequency Representation are as follows:

(i) Linear frequency range = 0.3Hz to 40Hz @ 0.5Hz step.

(ii) *n_cycles* for frequencies = (Linear frequency range) / 2.

(iii) Use fft based convolution.

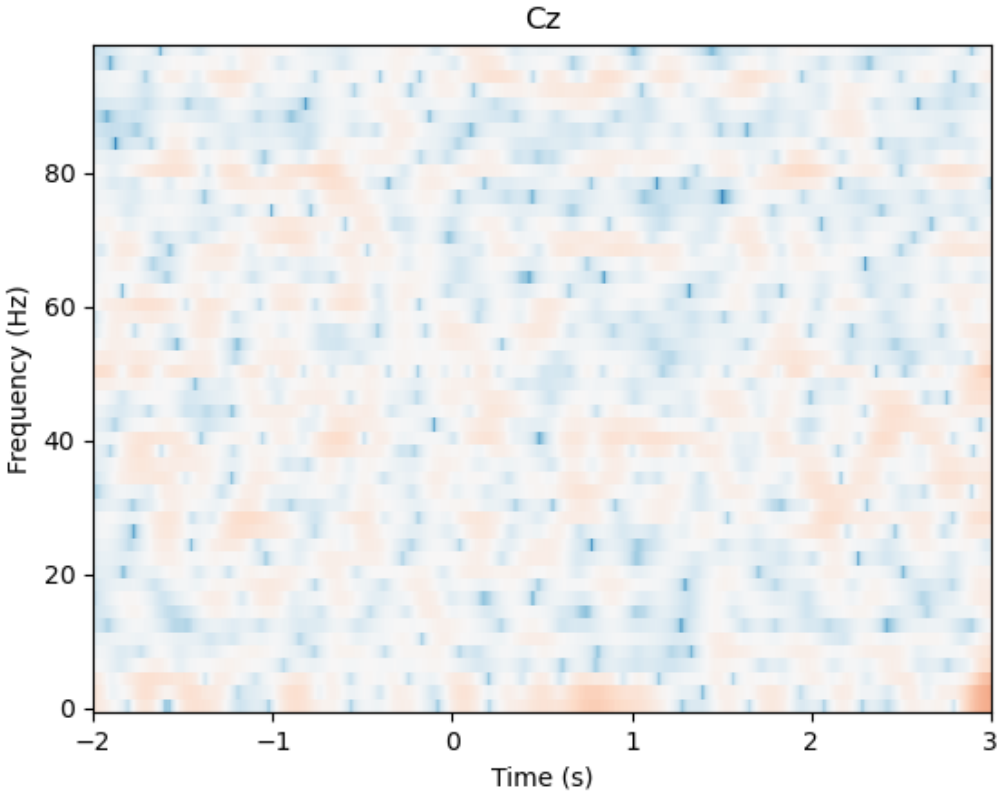


Figure 3.20: Time-Frequency Representation for Cz of single trial.

3.4.2 Conversion of TFR data to Data for CNN-LSTM Network

Data generation consists of multiple functional tasks. These functional tasks include Data loader, Sliding Window, Stratified Train Test Split, and Data Generator for CNN-LSTM Network.

3.4.2.1 DataLoader

This functional block loads TFR data of all channels and rearranges it into 2D array according to spatial information of each channel as shown in Table 3.4. Empty cells in the 2D array are filled with zeroes.

Table 3.4: Rearrangement of TFR data of EEG channels according to spatial information

	Col 1	**Col 2**	**Col 3**	**Col 4**	**Col 5**	**Col 6**	**Col 7**
Row 1				AFz			
Row 2		F3	F1	Fz	F2	F4	
Row 3	FFC5h	FFC3h	FFC1h	FFC2h	FFC4h	FFC6h	
Row 4	FC5	FC3	FC1	FCz	FC2	FC4	FC6
Row 5	FCC5h	FCC3h	FCC1h	FCC2h	FCC4h	FCC6h	
Row 6	C5	C3	C1	Cz	C2	C4	C6
Row 7	CCP5h	CCP3h	CCP1h	CCP2h	CCP4h	CCP6h	
Row 8	CP5	CP3	CP1	CPZ	CP2	CP4	CP6
Row 9	CPP5h	CPP3h	CPP1h	CPP2h	CPP4h	CPP6h	
Row 10	P5	P3	P1	Pz	P2	P4	P6
Row 11			PPO1h	PPO2h			
Row 12				POz			

3.4.2.2 Sliding Window

Each localized TFR data is divided into windows by using sliding window technique as shown in Figure 3.21. This step helps mitigate the lower classification accuracies caused by the non-stationarity problem seen in EEG classification. Settings for sliding windows generation are listed below:

(i) Start time = 2000ms from start of trial (When cue is shown)

(ii) Window size = 1000ms

(iii) Window step = 10ms

(iv) Number of consecutive windows = 20

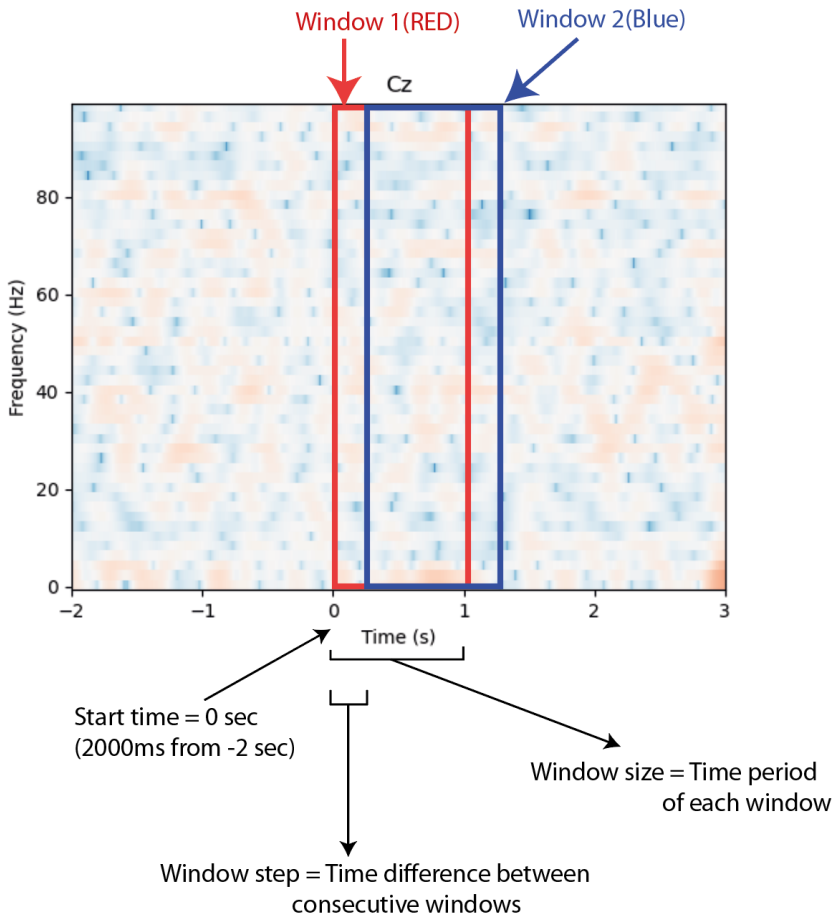


Figure 3.21: Parameters used in sliding window generation.

3.4.2.3 Stratified Train Test Split

Each window of localized TFR data is a sample data that can be fed to hybrid CNN-LSTM network. But before feeding the data, all the samples are split into training and testing datasets. A single training data consists of (X,y), where X= input data (window of Localized TFR), and Y = One hot encoding of the corresponding cue for input data. For the test set, a single test data consists of (X), where X = input data (window of Localized TFR). Each split is shuffled and stratified using attempted cues of the data. Training and testing sets contain samples of all classes according to split ratio. The split ratio for testing is set to 0.9, which implies each test set contains 90% of the data of all classes while the training set contains 10% data of all classes.

3.4.2.4 Datagenerator for CNN-LSTM Network

The main function of the Datagenerator function is to provide CNN-LSTM with a random sample of training data and test data. This bundles sample data in batches which are then fed to the CNN-LSTM network.

3.5 Classification

Classification is done using Deep Neural Network comprising of CNN and LSTM networks. Hybrid CNN-LSTM is used to extract spectral, temporal, and spatial features, and then classify them into five different classes. Initial layers of DNN are designed to extract useful feature informations. Later layers of DNN are used to classify features into one of five output classes. During the training phase, DNN learns relevant information about features to change the values of the parameters of the network. During the testing phase, the output is acquired for the test set using the learnt values of these parameters.

3.5.1 Deep Neural Network (DNN)

DNN makes use of a certain architectural arrangement of simple nodes that perform simple mathematical calculations. As a result, DNN performs complex tasks when considered as a black box. The parameters of each node are adjusted iteratively using Back Propagation Algorithm. Creating DNN architecture by arranging nodes into multi-layered architecture enables feature extraction and classification.

3.5.1.1 Conv3D

3 Dimensional Convolutional layer creates a convolution kernel that is convolved with the layer input to produce a tensor of outputs. A bias vector is created and added to the outputs. The values of 3D kernel are adjusted such as to minimize error at the output of the layer during backpropagation.

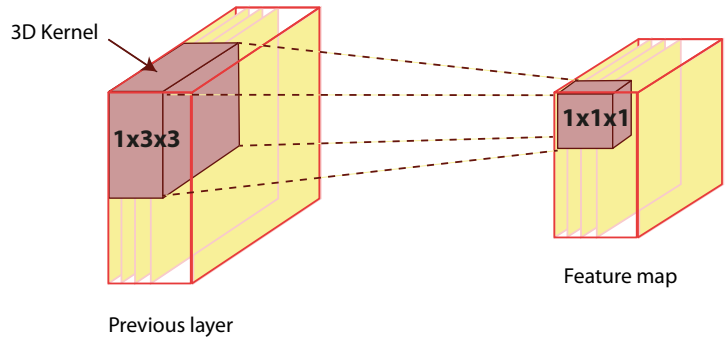


Figure 3.22: Three dimentional convolution operation.

Arguments of 3D Convolution layer are as follows:

(i) **Filters**: Defines the dimension of the output space (number of output filters created).

(ii) **Kernel size**: Defines the dimension of Kernel i.e. 3D convolution window.

(iii) **Strides**: Defines the strides of the convolution along all three dimension.

(iv) **Activation**: Activation function of 'tanh' is used

Input shape Input shape is a 5D tensor with shape: (batch, time_steps, localized_row, localized_col, freq_components)

Output shape A tensor of rank 5 representing activation(conv3d(inputs, kernel) + bias). Output is 5D tensor with shape: (batch, time_steps, localized_row, localized_col, no_of_filters)

3.5.1.2 Batch Normalization

Batch normalization layer returns normalized inputs. A transformation is applied where the output mean is nearly 0 and the standard deviation is nearly 1. Batch normalization runs differently during training and inference phase [102]. In training phase, the mean and standard deviation of the current batch is used to normalize the inputs. For each EEG channel, this layer returns the normalized values given by following Equation (3.7),

$$\frac{(x-\mu)}{\sqrt{\sigma^2}} \times \gamma + \beta \tag{3.7}$$

where,

- x is individual input values of each batch
- μ is mean of each batch
- σ is a variance across each batch
- γ is a learned scaling factor (initialized as 1)
- β is a learned offset factor (initialized as 0)

In inference phase, the layer uses moving average of mean and standard deviation to normalize the batches shown by Equation (3.8).

$$\frac{(x-\mu_i)}{\sqrt{\sigma_i^2}} \times \gamma + \beta \tag{3.8}$$

where,

- x is individual input values of each batch
- μ_i is moving mean in consecutive batches
- σ_i is a moving variance in consecutive batches
- γ is a learned scaling factor
- β is a learned offset factor

Moving mean and moving variance are updated each time the layer in called in inference mode as shown by Equation (3.9). Moving mean and moving variance are non-trainable parameters.

$$\begin{aligned} \mu_i &= \mu_{i-1} \times momentum + \mu \times (1 - momentum) \\ \sigma_i &= \sigma_{i-1} \times momentum + \sigma \times (1 - momentum) \end{aligned} \tag{3.9}$$

- μ_i is moving mean in current batch
- μ_{i-1} is moving mean in previous batch
- σ_i is moving variance in current batch
- σ_{i-1} is moving variance in previous batch

3.5.1.3 Average Pooling 3D

Average pooling 3D downsamples the input along 3 dimensions by taking the average value. An average value is returned for a 3 dimensional window over input data defined by its pool size. The window is shifted by strides along each dimension. Argument used for Average Pooling 3D is:

- **Pool size**: Defines factors by which downscale is performed in 3 dimensions (dim1, dim2, dim3)

3.5.1.4 ConvLSTM2D

Long Short-Term Memory (LSTM) To learn sequential features of time series data, LSTM is used. LSTM is a special Recurrent Neural Network (RNN) which is more stable for modeling long term sequence dependencies. LSTM uses memory cell c_t which acts as an accumulator of state information of the sequence. Different self parameterized controlling gates access, write, and clear the cell state. Information is accumulated in the cell state only if the input gate is activated. Also, there is a forget gate which clears past cell status c_{t-1} if the forget gate f_t is activated. The output gate o_t controls whether the output of cell state c_t will be forwarded to the final hidden state h_t. Figure 3.23 shows LSTM architecture. The advantage of the memory cell states and the control gates is to control information flow so that the gradient is confined in the cell. This prevents from vanishing gradient problem. The key equations of LSTM are shown in Equation (3.10), where '$\circ$' denotes the Hadamard product. In practice, multiple LSTM layers can be stacked in solving complex sequence modelling

applications.

$$
\begin{aligned}
i_t &= \sigma(W_{xi}\, x_t + W_{hi}\, h_{t-1} + W_{ci}\, c_{t-1} + b_i) \\
f_t &= \sigma(W_{xf}\, x_t + W_{hf}\, h_{t-1} + W_{cf}\, c_{t-1} + b_i) \\
c_t &= f_t \circ c_{t-1} + i_t \circ tanh(W_{xc}\, x_t + W_{hc}\, h_{t-1} + b_c) \\
o_t &= \sigma(W_{xo}\, x_t + W_{ho}\, h_{t-1} + W_{co} \circ c_t + b_o)
\end{aligned}
\tag{3.10}
$$

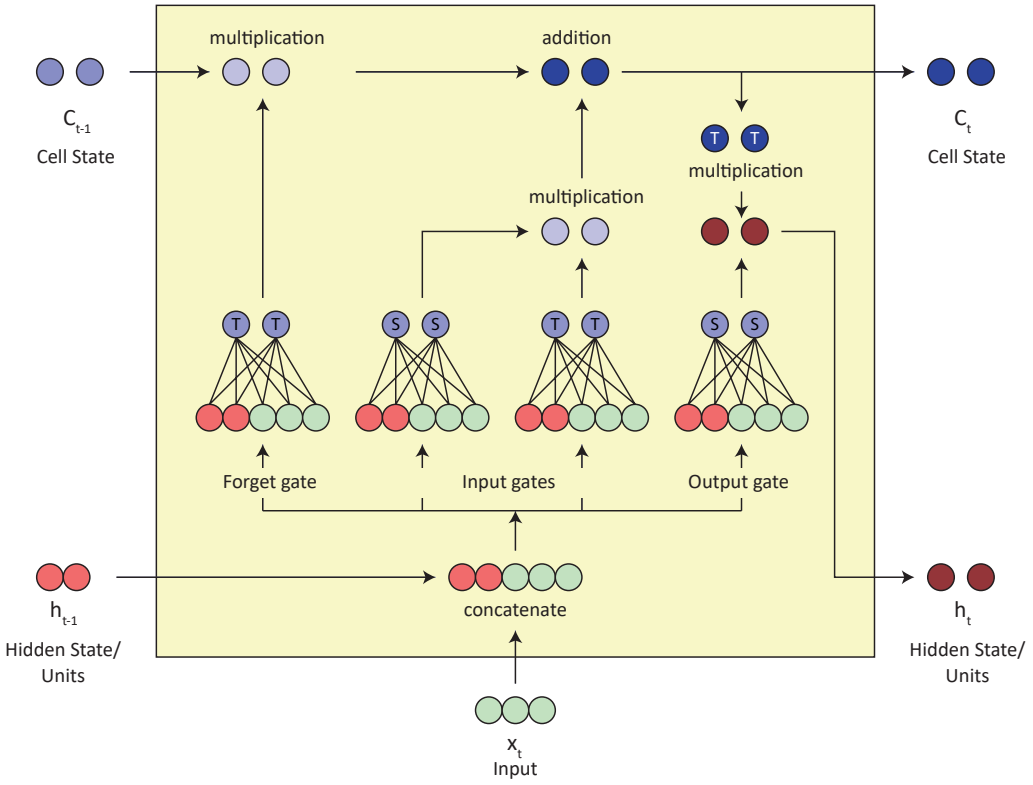


Figure 3.23: Architecture of LSTM network.

2D Convolutional LSTM Fully Connected LSTM performs very well in handling temporal correlation. In the case of spatial data, it contains too much redundancy. A convolutional structure is used in input-to-state and state-to-state transition to address this problem coined as 2D convolutional LSTM [103] as shown in Figure 3.24.

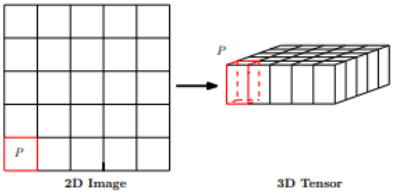


Figure 1: Transforming 2D image into 3D tensor

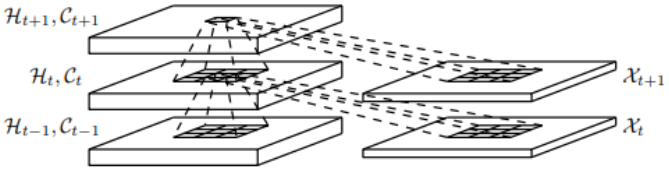


Figure 2: Inner structure of ConvLSTM

Figure 3.24: Architecture of 2D convolutional LSTM network.

The most important drawback of fully connected LSTM is that inputs should be reshaped to a single dimension. During reshaping, all spatiotemporal information is lost. To overcome this problem, all the inputs $X_1, ..., X_t$, cell outputs $C_1, ..., C_t$, hidden states $H_1, ..., H_t$, and gates $i_t, f_t,$ *and* o_t of the ConvLSTM are converted to 3D tensors. The first dimension is the temporal dimension while the last two dimensions are spatial dimensions (i.e. row and column). The inputs and past states are used to determine the future state of a certain cell. For each value in a state, the past state of its local neighbours is used. This can be achieved by using a convolution operation in the state-to-state and input-to-state transitions as shown in Figure 3.24 (Figure 2). The key equations of ConvLSTM are shown in Equation (3.10) where gates are 2 dimensional for a specific time frame. Arguments for ConvLSTM2D:

- **Filters**: Number of output filters in the convolution
- **Kernel size**: The dimension of the 2 dimensional convolution window.
- **Return sequences**: If true full sequence is returned.

3.5.1.5 Dropout

A random set of input units are set to 0 by the dropout layer. The inputs are set to 0 according to frequency of the dropout rate during the training phase. This helps prevent the overfitting problem. Input values which are not set to 0 are then scaled up by $\frac{1}{1-dropoutrate}$. Scaling is done so that sum over all inputs is unchanged. It is

to be noted that the Dropout layer is active only in training mode. Arguments of dropout:

- **Rate**: Defines fraction of the input units to be dropped.

The dropout rate is set to 0.25 for the ConvLSTM2D regularization.

3.5.1.6 Dense Layer

Dense layer is a layer where every input node is connected to every output node. At every output node, it implements a simple operation of input multiplied by a certain weight and added with a bias term. The operation of dense layer can be shown by the equation below:

$$output = activation(dot(input, kernel) + bias)$$

where activation is the element-wise activation function
kernel is the weights matrix
bias is a bias vector
Arguments:

- **Units**: Defines the number of dense layer nodes and dimension of the output tensor.
- **Activation**: Activation function of to be used. Function 'tanh' is used in the dense layer.

3.5.1.7 Softmax Activation Function

Softmax function is a function which converts a vector of values to equivalent probability distribution. The feature of the softmax function is that its output ranges from 0 to 1 for any input values of input and the sum of all the output equals unity. Softmax is used as the activation for the last layer of a classification network.

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}} \tag{3.11}$$

Where, $i = 1, \ldots, K \; and \; z = (z_1, \ldots, z_k)$

3.5.1.8 Binary Cross Entropy Functions

Binary Cross Entropy function is the negative average of the log of corrected predicted probabilities. It compares predicted probabilities to actual class output hence corrected probabilities. Binary cross entropy uses the corrected predicted probabilities to calculate a loss score based on penalizing according to the deviation from the expected value. This function is used as a loss function in the DNN.

$$logloss = -\frac{1}{N}\sum_{i}^{N}\sum_{j}^{M} y_{ij} \; log(p_{ij}) \tag{3.12}$$

Where, N = number of rows M = number of classes

3.5.1.9 Adaptive Momentum Optimization (ADAM)

ADAM is a stochastic gradient descent algorithm which is based on adaptive estimation of first-order and second-order moments. ADAM optimizer is computationally efficient, requires less memory, is not affected by diagonal rescaling of gradients, and is well suited for big data and a huge number of parameters [104]. The mathematical representation for first order moment and second order moment is given in Equation (3.13).

$$\begin{aligned} V_{dw} &= \beta_1 \times V_{dw} + (1 - \beta_1) \; dW \\ V_{db} &= \beta_1 \times V_{db} + (1 - \beta_1) \; db \\ S_{dw} &= \beta_2 \times S_{dw} + (1 - \beta_2) \; dW^2 \\ S_{db} &= \beta_2 \times S_{db} + (1 - \beta_1) \; db^2 \end{aligned} \tag{3.13}$$

Where, V_{dw} is first order moment or moving average of gradient of weight parameter

β_1 is first order moment constant (default value = 0.9)

dw is gradient of loss function with respect to weight

V_{db} is first order moment or moving average of gradient of bias parameter

db is gradient of loss function with respect to bias

S_{dw} is second order moment or moving average of square of gradient of weight parameter

β_2 is second order moment constant (default value = 0.999)

S_{db} is second order moment or moving average of square of gradient of bias parameter

Then, the correction in network parameters (weights and bias) in each iteration during the training process is represented in Equation (3.14). This will in turn gradually improve the performance of the model in successive iterations.

$$\begin{aligned} weight_{new} &= weight_{old} - \alpha \times \frac{V_{dw}}{\sqrt{S_{dw} + \epsilon}} \\ bias_{new} &= weight_{old} - \alpha \times \frac{V_{db}}{\sqrt{S_{db} + \epsilon}} \end{aligned} \tag{3.14}$$

3.5.1.10 Metrics

Various metrics are used to measure the classification performance of EEG classification using hybrid CNN-LSTM network. The metrics used in this research include: categorical accuracy, confusion matrix, precision, recall, F1 score, and Cohen's Kappa.

Categorical Accuracy Categorical Accuracy is the ratio of the total number of correctly classified predicted values to the total number of predicted values. The total number of predicted values includes correctly classified values along with incorrectly classified values.

Confusion Matrix Confusion matrix is a table layout for the visualization of errors in the classification task. One dimension of the matrix represents the actual class while the other represents the predicted class. The total number of classified samples is written in cells corresponding to their original class and predicted class. Confusion

matrix is made up of four metrics:

(i) **True Positive**: It is an instance when positive samples are correctly classified as positive.

(ii) **True Negative**: It is an instance when negative samples are correctly classified as negative.

(iii) **False Positive**: It is an instance when negative samples are incorrectly classified as positive.

(iv) **False Negative**: It is an instance when positive samples are incorrectly classified as negative.

Figure 3.25 shows the distribution of true positive, true negative, false positive, and false negative in the confusion matrix.

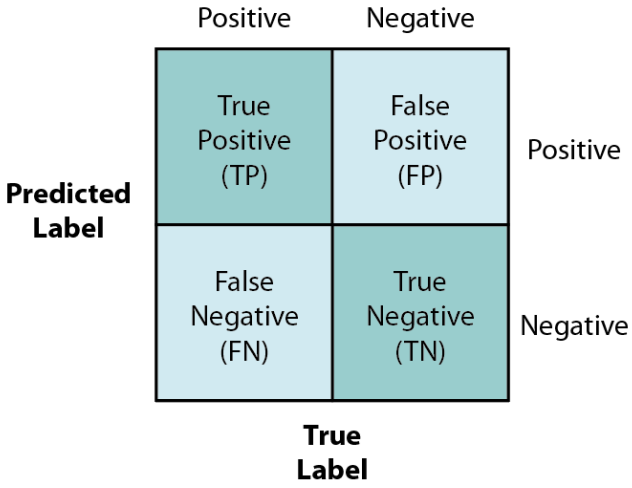


Figure 3.25: Confusion matrix.

Precision Precision is defined as the ratio of True Positive to the total number of samples classified as positive.

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \tag{3.15}$$

Recall Recall is defined as the ratio of True Positive to the total number of samples which are originally true. Recall measures the ability to detect positive samples.

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{3.16}$$

F1 score The accuracy metric might be misleading if the dataset is imbalanced (each class of the dataset does not have the same number of samples). F1 score combines precision and recall using harmonic mean. Maximum value of F1 score is achieved if both precision and recall are high.

$$F1\ score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{3.17}$$

Cohen's Kappa Cohen's kappa coefficient (κ) is a statistical tool used to measure inter-rater and intra-rater reliability for categorical classification [105]. It takes into account the possibility of the agreement occurring by chance given by $\frac{1}{Number\ of\ output\ classes}$. This metric measures the agreement between two raters who each classify the same number of items into the same number of mutually exclusive categories.

$$\kappa = \frac{p_o - p_e}{1 - p_e} \tag{3.18}$$

where p_o is the relative observed agreement among raters,
p_e is the hypothetical probability of chance agreement, using the observed data to calculate the probabilities of each observer randomly seeing each category.

3.5.2 CNN-LSTM Network Architecture

To achieve the objective of classifying EEG signals, three main features of EEG signals are to be considered. These features include:

- Frequency component of each channel.
- Spatial information of each frequency component.
- Time series variation of spatial information.

Thus, a CNN-LSTM architecture is constructed as shown by Figure 3.26. CNN-LSTM architecture consists of Frequency learning Convolutional Layer, Time series reduction Convolutional Layer, Spatial information learning Convolutional layer, Time series information learning Convolutional LSTM, and finally end classifier CNN. A combination of Conv3D layer, Batch normalization layer, AveragePooling3D layer, ConvLSTM2D layer, Dropout layer, Dense layer, and Softmax activation function is used to construct detailed Deep Neural Network hybrid CNN-LSTM classifier.

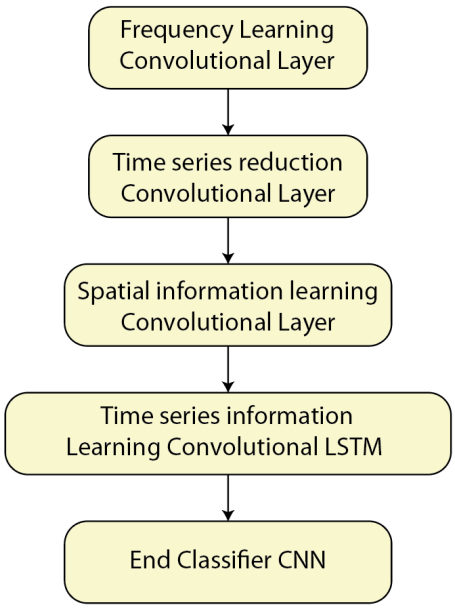


Figure 3.26: Hybrid CNN-LSTM architecture.

3.5.2.1 Frequency Learning Convolutional Layer

CNN-LSTM network Data generator provides CNN-LSTM network with batches of data of shape (batch, time steps, localized row, localized col, number of frequencies). Thus, the actual shape of input data becomes (32, 80, 12, 7, 40). There are 40 frequency components from TFR data. Localized rows and localized columns carry the spatial information of EEG data. To learn those frequency components Conv3D is used with 16 filters. This means that 40 frequency nodes are reduced to 16 nodes. It is to be noticed that Kernel (1,1,1) is used which only affects frequency information. Batch normalization is used to normalize the inputs to the next layer.

3.5.2.2 Time Series Reduction Convolutional Layer

Time series reduction is done to reduce time series information while concurrently learning the most effective values of the kernel to be used for the reduction. This layer is equivalent to the weighted sum of frequency information and time series information obtained from the previous layer. The weights of the weighted sum are kernel values. Conv3D layer concurrently reduces 16 frequency information nodes from the previous layer and 5 time series information to the same 8 nodes. These 8 nodes contain frequency information in reduced time steps. Also, this layer reduces the number of time series data for efficient learning by ConvLSTM2D layer used later in CNN-LSTM network. Conv3D with kernel (5,1,1) is used along with AveragePooling3D layer. AveragePooling3D again reduces obtained time series values by a factor of two.

3.5.2.3 Spatial Information Learning Convolutional Layer

After time series information is reduced, localization information is learnt by Conv3D with kernel (1,3,3). This shows that convolution kernel (1,3,3) is applied to (time steps, localized row, localized col) while also connecting 8 time series reduction nodes to 8 spatial information nodes. This in turn learns the combined effect of spatial

information along with features learnt in previous layers.

3.5.2.4 Time Series Information Learning Convolutional LSTM Layers

All the learnt information contain time series variation, which is learnt by two stacked ConvLSTM2D layer. Each ConvLSTM2D performs convolution operation before learning time series information using LSTM modified to work with 2D filters. 64 filters are used for each ConvLSTM2D to learn time series information. Dropout is required for regularization purpose. All sequence is returned from ConvLSTM2D to use this information for the final classification process.

3.5.2.5 End Classifier CNN

The generated sequence which is provided by ConvLSTM2D layer is then passed to End Classifier CNN to finally get classification results. For this purpose, two layers of Conv3D for the reduction of nodes are used before flattening and performing the final classification by Dense layer. Softmax function is used in the last layer to give us probabilities of classification. Detailed architecture of the proposed hybrid CNN-LSTM architecture is shown in Figure 3.27. Optimizer, Loss function, metrics, and other parameters are as follows:

(i) Optimizer: ADAM (Learning rate = 0.001)

(ii) Loss function: Binary cross entropy

(iii) Metric: Categorical accuracy

(iv) Maximum epochs: 100

(v) Early stopping: Monitor Validation Loss (Patience = 10)

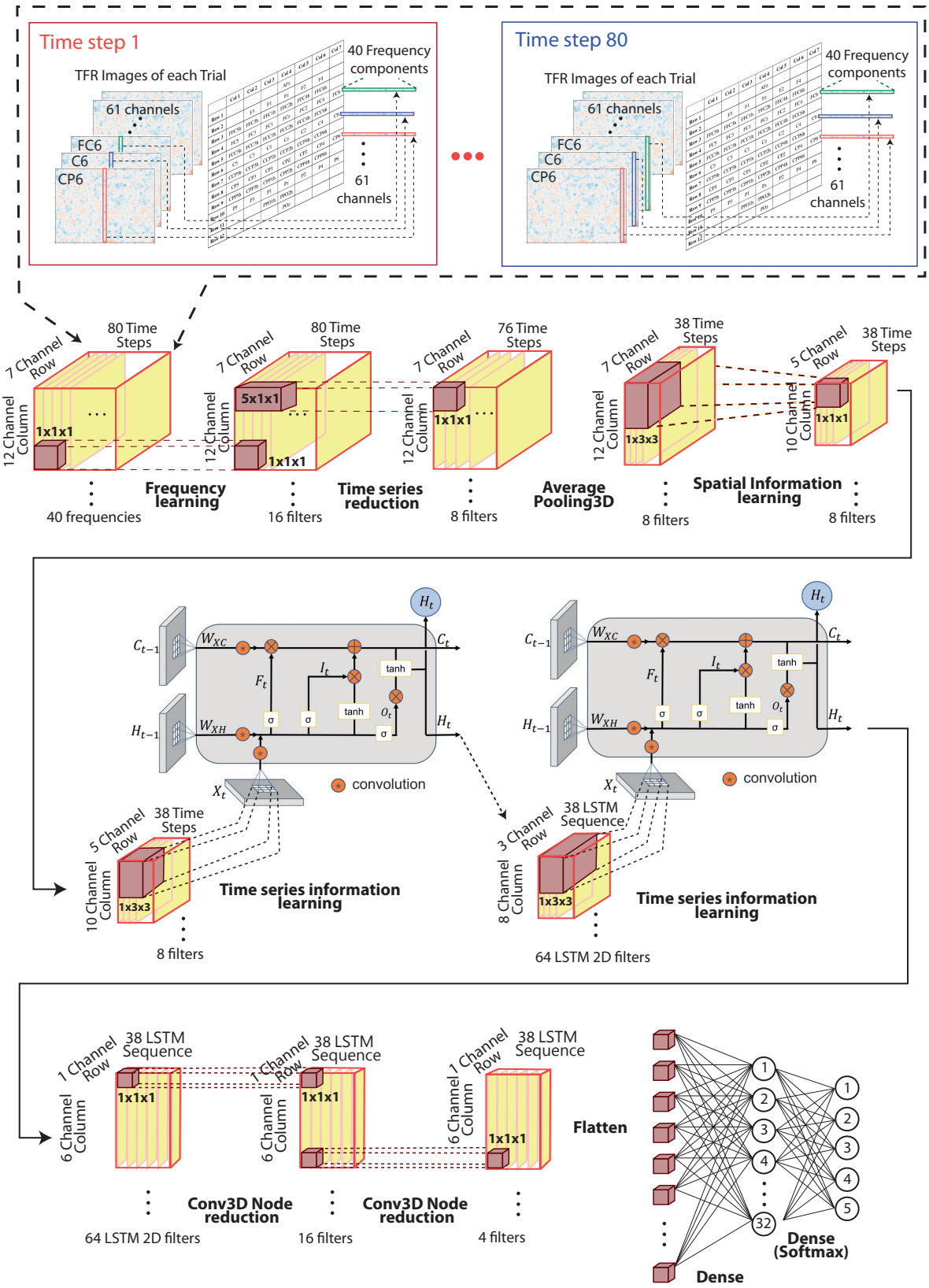


Figure 3.27: Hybrid CNN-LSTM network architecture.

3.6 Chapter Summary

EEG pre-processing, Post-processing, and Classification steps are used for five class classification of attempted hand movements of SCI patients. Filtering, downsampling, and artifact removal using ICA are done for EEG pre-processing. Time-Frequency Representation of EEG data along with Datagenerator is used for post processing step. Datagenerator applies spatial encoding, windowing technique, splits train and test sets and randomly selects data to feed hybrid CNN-LSTM network. CNN-LSTM network learns spectral features, spatial features, and temporal features from the training set. Learnt values of CNN-LSTM are finally used for inference which results in classification into 5 different classes.

CHAPTER 4
RESULTS AND DISCUSSIONS

4.1 Introduction

Proposed CNN-LSTM network architecture is used to classify five different classes of attempted hand movements of 10 SCI patients. In this chapter: evaluation criteria for EEG classification; analysis of the effect of various factors on performance of the classifier; obtained classification performance; and comparison of obtained results with literature is done.

4.2 Evaluation Criteria

For the evaluation of EEG classifier, following evaluation metrics are used:

(i) Categorical accuracy (Testing set categorical accuracy)

(ii) Loss value (Binary Cross Entropy loss function)

(iii) Confusion matrix

(iv) Precision

(v) Recall

(vi) F1 score

(vii) Cohen's Kappa value

Each window of EEG TFR data generated by Post-processing step is used to calculate the categorical accuracy. Testing set categorical accuracy shall be referred to as accuracy here forth for ease of understanding. The training set categorical accuracy is 99.5%±0.4% for all experiments unless specified. Two graphs are considered simultaneously for the evaluation of accuracy and generalizability of the classifier: the response of categorical accuracy with respect to the number of epochs; the response of loss value with respect to the number of epochs. Also, the response

of training data is seen along with the response of test data for accuracy and loss. A classifier can be considered to have good generalizability, if both the response of training and test data are similar. Figure 4.13 depicts good accuracy and generalizability of the classifier. Confusion matrix is used for detailed output results with respect to different classes. It is used for the calculation of various metrics like precision, recall, F1 score, and Cohen's kappa values.

4.3 Performance Analysis

Various aspects are considered for the optimization of CNN-LSTM model to achieve maximum classification performance. Effects of various factors are analyzed which includes: artifact removal, sliding window, and effect of train-test ratio.

4.3.1 Effects of Artifact Removal using ICA

Literature showed that various strategies of artifact removal in motor imagery classification have been applied (shown in Figure 2.3). These strategies include automatic removal, manual removal, artifact left in, and artifact not addressed. Thus, the classification performance of the classifier is seen with and without artifact removal using ICA. Five individual experiments are done for both artifact removal using ICA, and no artifact removal. Results show that artifact removal using ICA helps in increasing the average classification accuracy by 0.94%. This small increase in classification accuracy is seen because hybrid CNN-LSTM network is able to handle non-stationarity very well. In case of no removal of artifacts, CNN-LSTM network uses artifacts also to classify EEG tasks which might be undesirable. Figure 4.1 shows increase in accuracy with the use of artifact removal using ICA where,

- Colored box represents the range of values covered by Q1 through Q3 of the data.
- Horizontal line inside the box represents Q2 value of the data.

- Maximum of upper tail represents the maximum value of the data.
- Minimum of lower tail represents minimum value of the data.
- Cross represents the mean value of data.

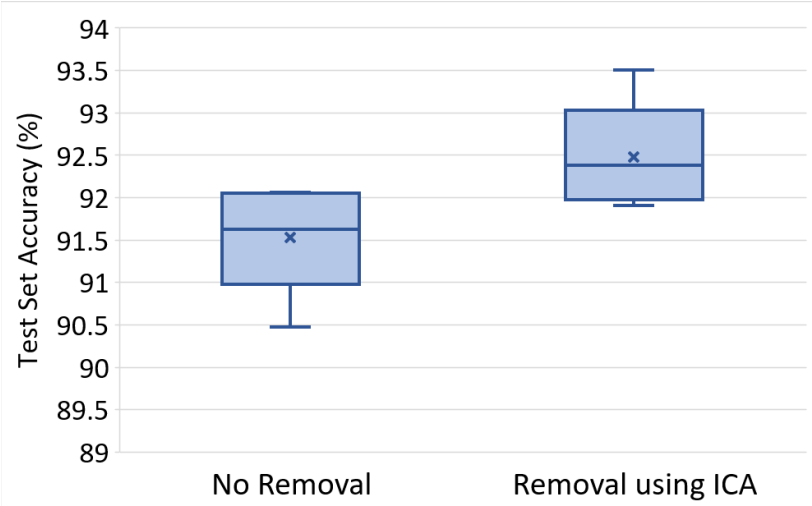


Figure 4.1: Box diagram of the effect of artifact removal on test set classification accuracy.

4.3.2 Effect of Sliding Window

For the optimization of parameters of Sliding windows: window step, window size and no of windows are analyzed. It is seen that 10ms window step yields the highest classification performance (see Figure 4.2). As the step size increases, the amount of missed information also increases causing a decrease in classification performance. As for window size 1000ms captures the optimum amount of information required for classification (see Figure 4.3). The classification performance saturated after 20 windows indicating that many windows are required to deal with the problem of non-stationarity in EEG classification (see Figure 4.4).

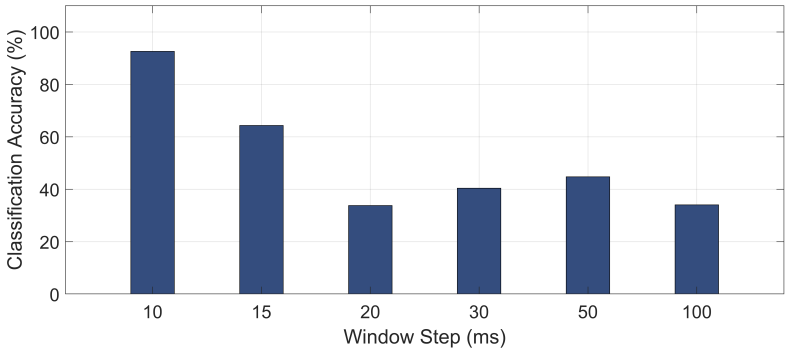


Figure 4.2: Effect of window step in accuracy.

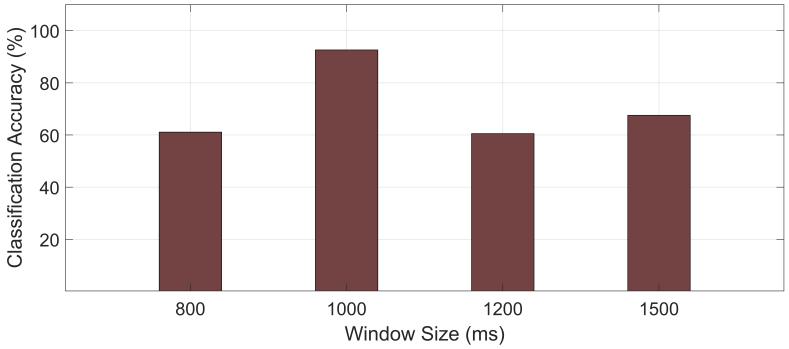


Figure 4.3: Effect of window size in accuracy.

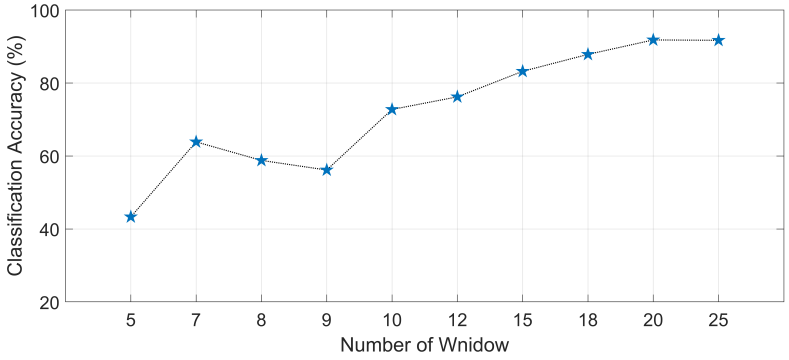


Figure 4.4: Effect of number of windows in accuracy.

4.3.3 Effect of Training and Testing Split

To test the validity of the CNN-LSTM network and its performance for various split ratios of training set to test set, test set accuracy and test set loss are observed for various split ratios for a patient. Here split ratio refers to the amount of test data relative to the whole data. Test set accuracy and loss are observed for a split ratio

ranging from 0.2 to 0.9. Figure 4.5 and 4.6 show the effect of split ratio on test set accuracy and loss. It can be seen that as the amount of test set increases and training set decreases, there is a decrease in final classification accuracy depicted by Figure 4.5. It should be noted that even for the split ratio of 0.9, i.e. 10% of the data is used for training and 90% of the data is used for testing purposes, the final achieved classification is 93%. This result is further validated by the final achieved loss value by hybrid CNN-LSTM network. Final achieved loss is more for the split ratio of 0.9 compared to others. Likewise, it can be seen that when the split ratio is 0.2, maximum classification performance is reached in a single epoch. It is the same for the split ratio of 0.3. As the split ratio increases gradually to 0.9, the number of epochs to obtain maximum classification performance also increases. This effect is also seen in Figure 4.6 graph, as there is a rapid decrease in loss value for split ratio 0.2 and the rate of decrease slows down as the split ratio increases.

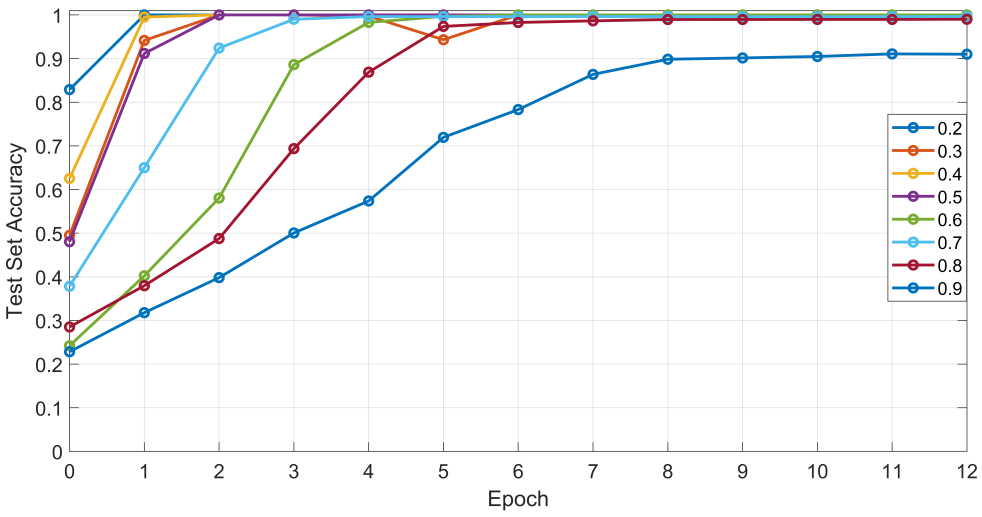


Figure 4.5: Effect of split ratio on test set accuracy.

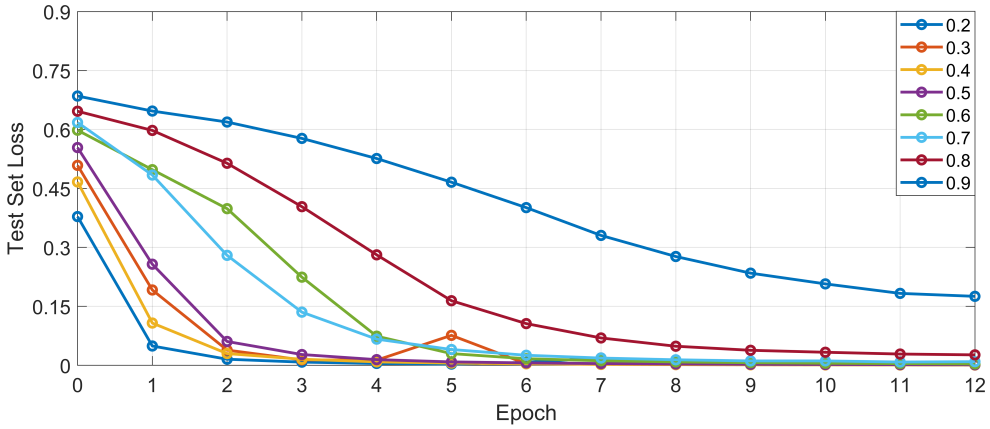


Figure 4.6: Effect of split ratio on test set loss.

4.3.4 CNN-LSTM Network Optimization

Various hyperparameters have been tuned to their optimum values for achieving maximum validation classification performance. Optimal values of the hyperparameters of hybrid CNN-LSTM network are as follows:

(i) Number of 3D Convolution filters for frequency learning = 16

(ii) Number of 3D Convolution filters for time series reduction filters = 8

(iii) Number of 3D Convolution filters for extraction of spatial information = 8

(iv) Number of ConvLSTM2D filters for sequence learning = 64

(v) Number of 3D Convolution filters for first layer reducing network = 16

(vi) Number of 3D Convolution filters for second layer reducing network = 4

(vii) Number of nodes in Classification Dense layer = 32

Multiple experiments have been carried out for the optimization of particular hyperparameters. Only the hyperparameter in test is varied while the remaining hyperparameters are kept constant. To evaluate the performance of the specific values of hyperparameters, two graphs are considered: Accuracy vs Epoch; Loss vs

Epoch. Each of these two graphs contains a plot of training set and test set of data for different hyperparameter values which are separated by their color.

For Accuracy vs Epoch graph, a gradual increment of the accuracy of training and test data is observed. Special attention should be given to the maximum accuracy value obtained by test set which is generally lower. The highest accuracy achieved by the test set is chosen. As for Loss vs Epoch graph, it can be seen that training loss shows a characteristic exponential decreasing curve. Here, special attention should be given to the response of loss of test set. Especially, whether the loss is decreasing or increasing, and by what amount. Considering these criteria, optimum values of hyperparameters are obtained. Figure 4.7, 4.8, 4.9, 4.10, 4.11, 4.12 shows response of network with various hyperparameter settings.

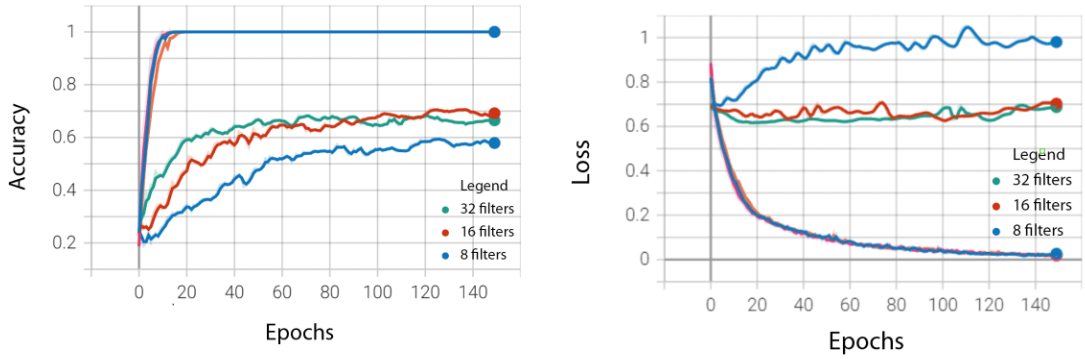


Figure 4.7: Frequency filter optimization.

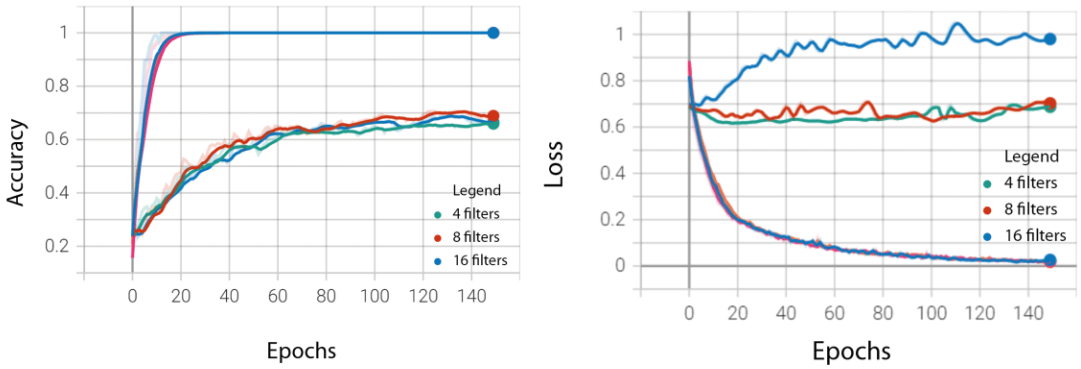


Figure 4.8: Time filter optimization.

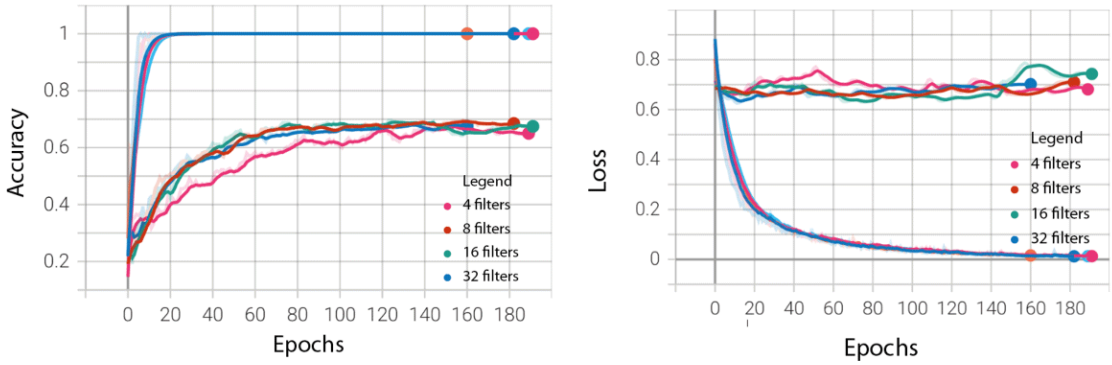


Figure 4.9: Localization filter optimization.

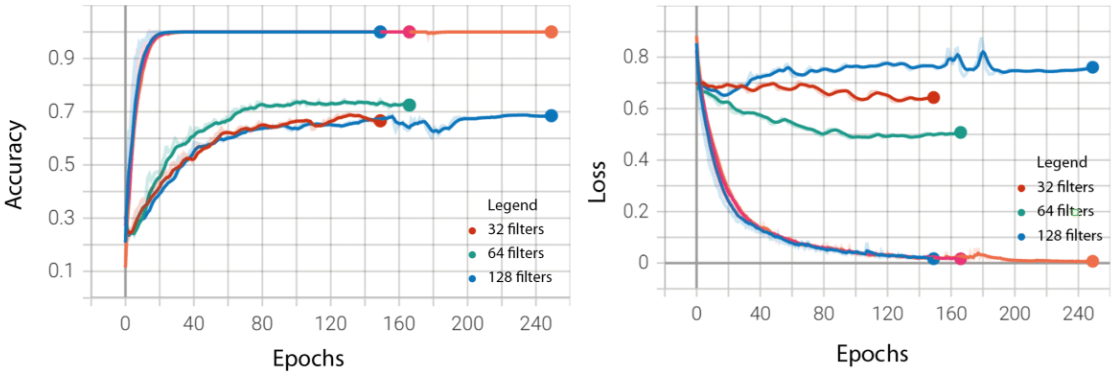


Figure 4.10: Convolutional LSTM filter optimization.

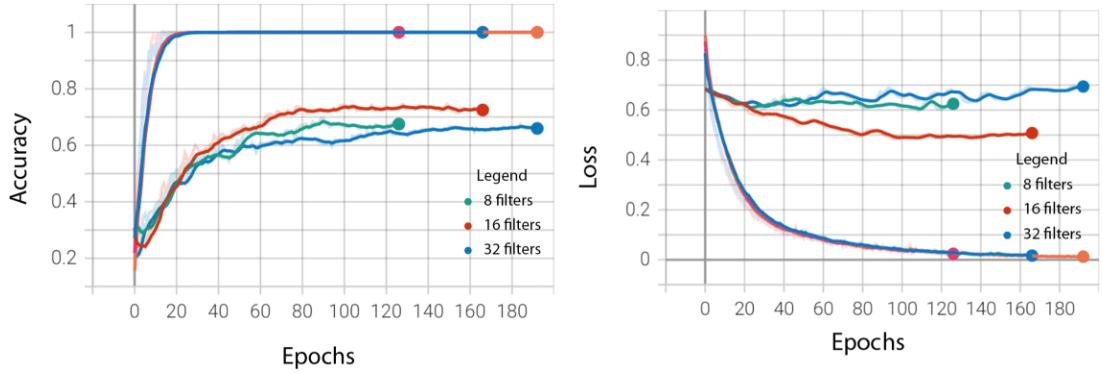


Figure 4.11: Node reduction filter optimization.

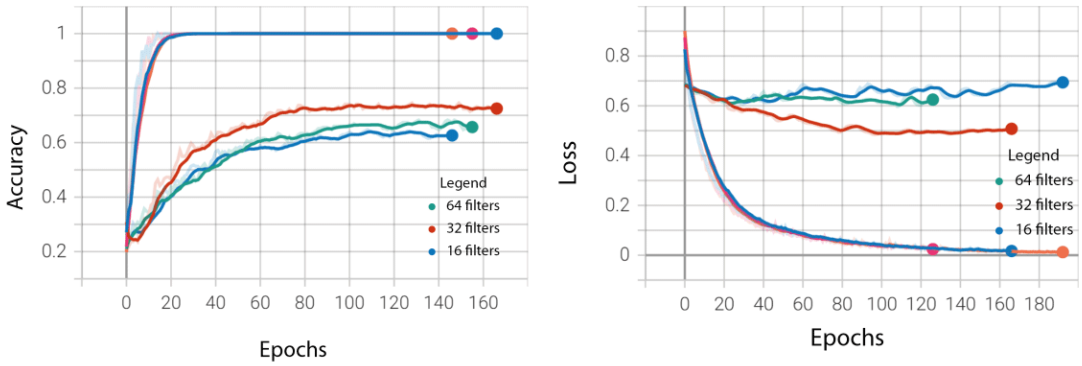


Figure 4.12: Dense layer nodes optimization

4.3.5 Computational Requirement of Classifier

The experiments have been carried out in a computer having Core i9-11980K and MSI RTX 3080 GPU. Hybrid CNN-LSTM DNN architecture had 4,98,185 parameters. The size of entire trained network is 5.84 MB (MegaBytes) excluding EEG data. It took an average of 16.1 seconds to complete each epoch. For each epoch, there are a total of 7200 windows of 61 channels of EEG TFR data. Each batch of EEG TFR data containing 32 samples took an average of 780 ms to complete during training. To achieve maximum classification performance with data of all runs of each patient, it took an average of 3.567 minutes at 10 epochs (with a split ratio of 0.9) excluding data loading time.

4.3.6 Final Classification Results

Figure 4.13 shows the learning curve of the network for Patient P01. Taking into account the calibration time issue in BCI, hybrid CNN˙LSTM classifier used 10% of the dataset for training to achieve maximum classification performance using less training data. This means the classifier is able to generalize using less amount of data. In this research, following results are achieved:

(i) Using only 10% of the EGG data of attempted movements of SCI patients for

training purposes, hybrid CNN-LSTM network is capable of learning features to classify 90% of the remaining data with a testing set accuracy of 92.36%±1.5% and training accuracy of 99.5±0.4%.

(ii) Proposed EEG classification procedure is able to use and extract spectral, temporal, and spatial features from high dimensional EEG data (e.g. 61 EEG electrodes used in this research).

It shows that the model is capable of generalizing the system by training on 10% of the dataset. Also, Figure 4.14 shows that the loss of the training and testing dataset decreased in similar manner further supporting the good generalization of the classifier and validating the performance of the network.

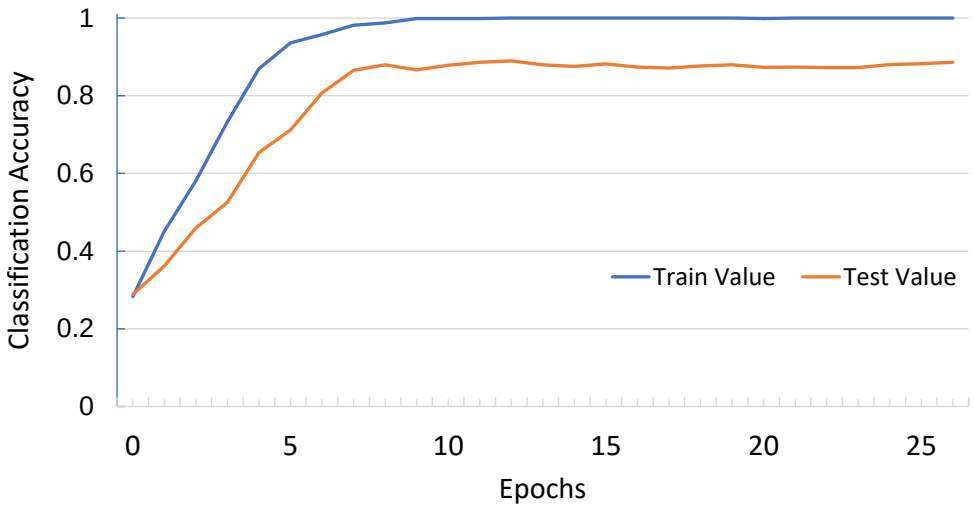


Figure 4.13: Accuracy-Epoch Plot for Patient 01.

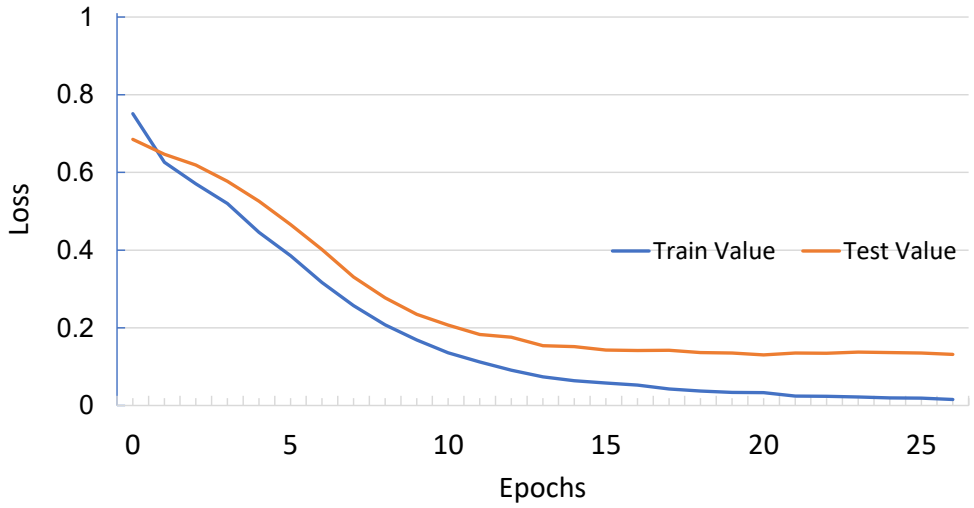


Figure 4.14: Loss-Epoch plot for Patient 01.

10 separate experiments with random sampling of data are carried out each yielding its own accuracy and loss (e.g. Figure 4.13 and 4.14). Obtained results from 10 experiments for each patient are used to acquire the distribution of classification accuracy shown by Figure 4.15.

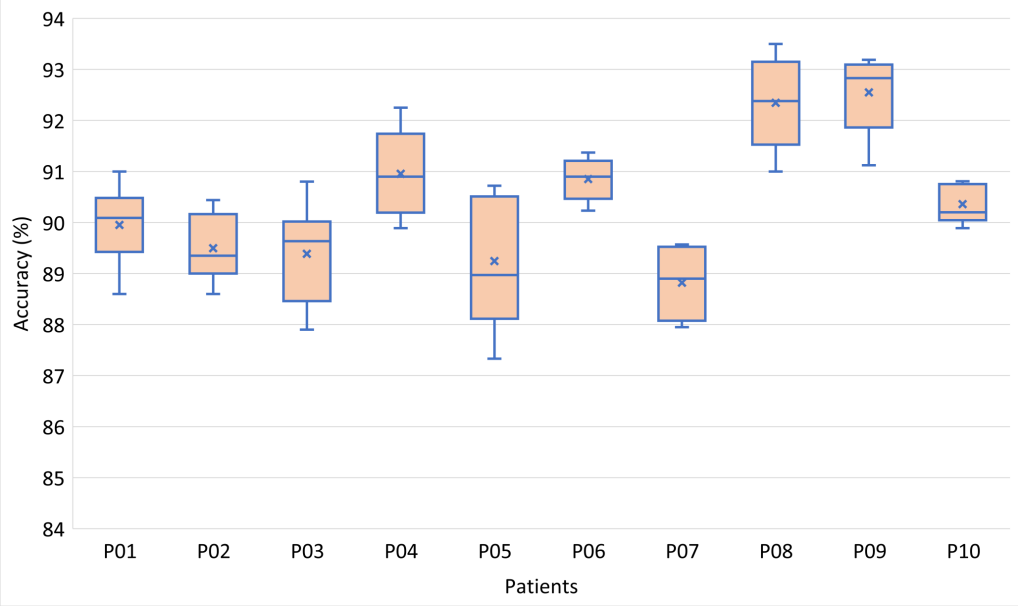


Figure 4.15: Box plot of classification accuracy for all patients.

4.3.6.1 Confusion Matrix of classification results

Figure 4.16 shows the confusion matrix for 5 class classification of attempted hand movements for patient P08. This shows good classification performance and generalization for all the classes. For a single trial, multiple windows are compared and the output class is chosen according to maximum probability given by hybrid CNN-LSTM network across all windows of this trial. This increases the overall accuracy of the classifier by an average of 1.96%.

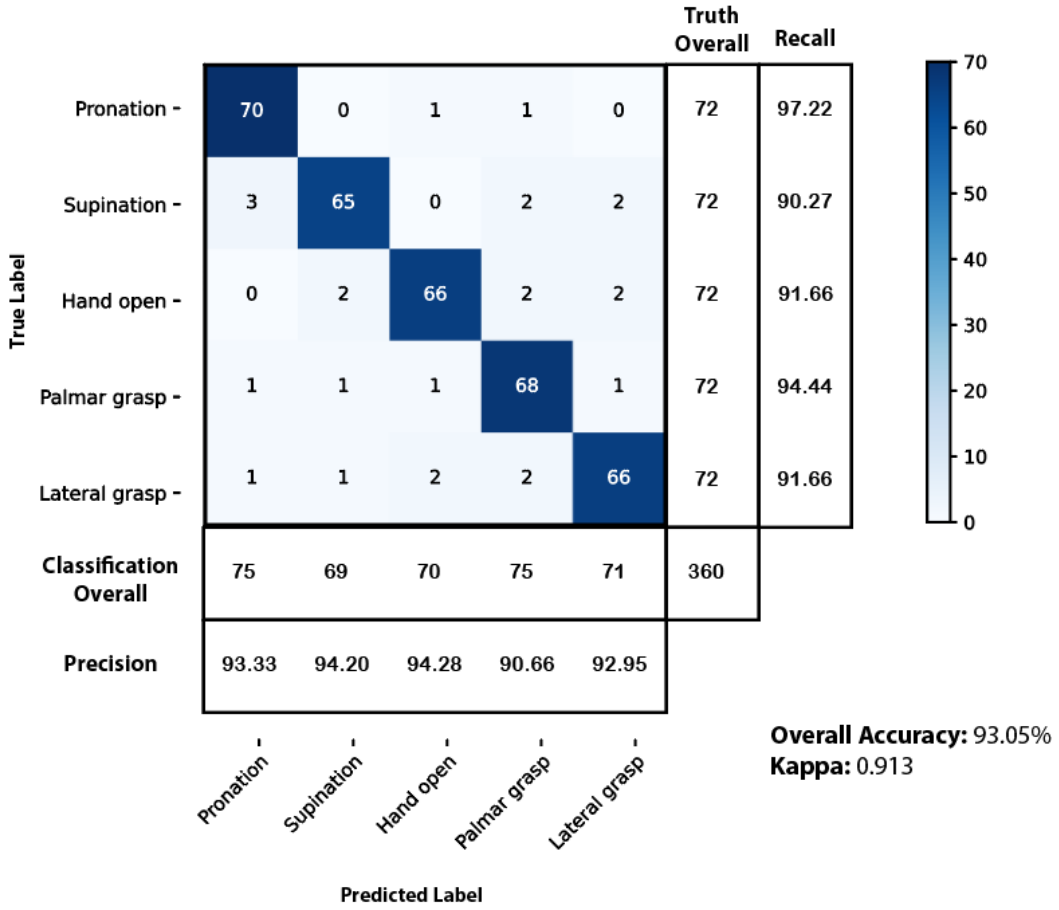


Figure 4.16: Confusion matrix for classification of 360 trials of Patient P08.

Using classification results of all patients, a confusion matrix is constructed which sums up classification results for all patients shown by Figure 4.17. The overall classification performance obtained is as follows:

(i) Overall accuracy: 92.363

(ii) Cohen's κ Value: 0.905

(iii) Precision: 92.384

(iv) Recall: 92.364

(v) F1 Score: 92.37

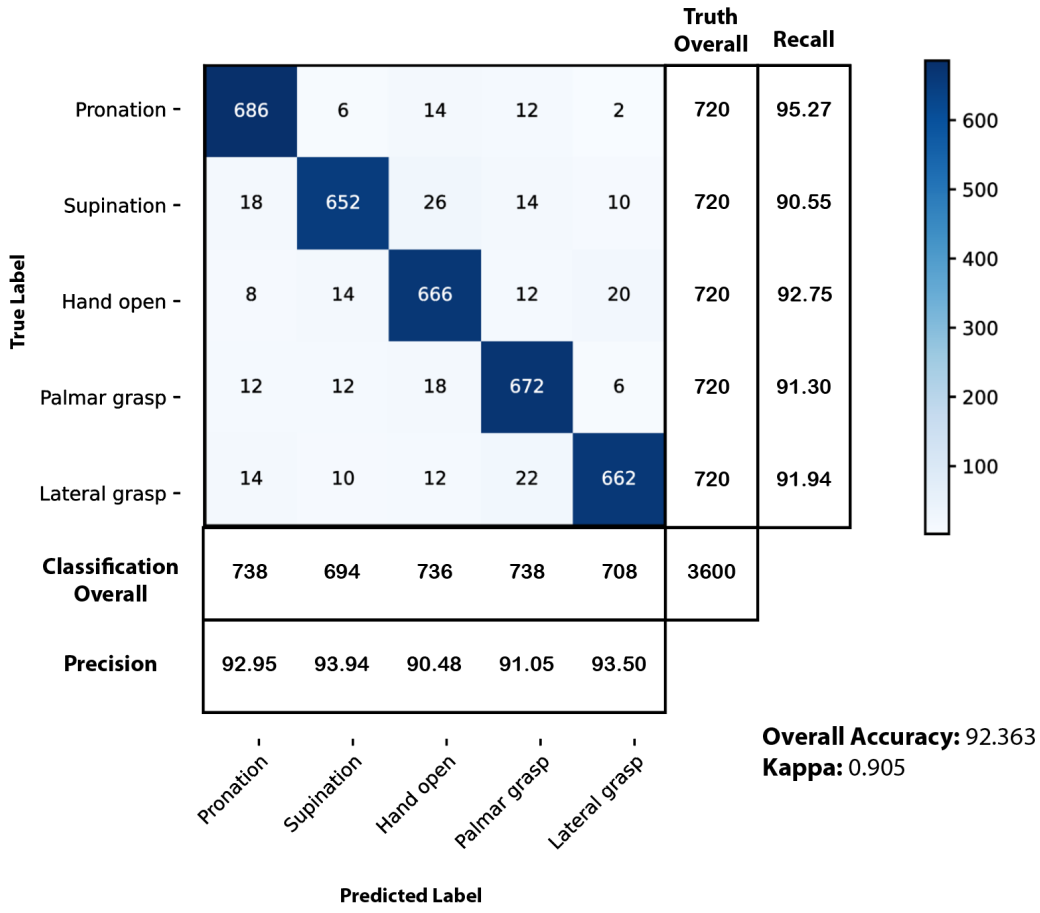


Figure 4.17: Confusion matrix for all patients.

4.3.7 Comparative Results

Table 2.4 shows various related EEG classification studies according to their output class, strategies, tasks, subject, training set, test set, and accuracy. It is to be noted that all the studies except for the task of attempted hand movements show that they classify certain combinations of left hand, right hand, tongue, feet, and eye closing tasks. Meanwhile, attempted hand movements deal with various movements from a single hand. These hand movements originate from a smaller region of the brain compared to other movements. This increases the complexity in classification because

of additional interference among signals which makes it difficult to separate brain response from artifacts. Further adding to the complexity, there is a greater extent of non-stationarity and high chances of artifacts in EEG signals acquired from SCI patients. The results of this research show that it achieves a consistent classification performance across all patients which is not the case with related literature. This is an increase of 47.36% classification accuracy compared to [20] for 5 class classification of attempted movement of SCI patients for the same dataset. This indicates that EEG classification process according to this research is able to handle non-stationarity of EEG signals and is flexible to handle a wide variety of data by extracting relevant features and classifying them.

Figure 4.18 shows a graphical comparison of this research among related EEG classification studies. This research shows an increase in classification performance across related literature for EEG classification of 5 classes while using 10% of the dataset for training purposes. This shows CNN-LSTM model is generalizing the data better than related studies while dealing with more complex scenarios of single hand movement and five output classes.

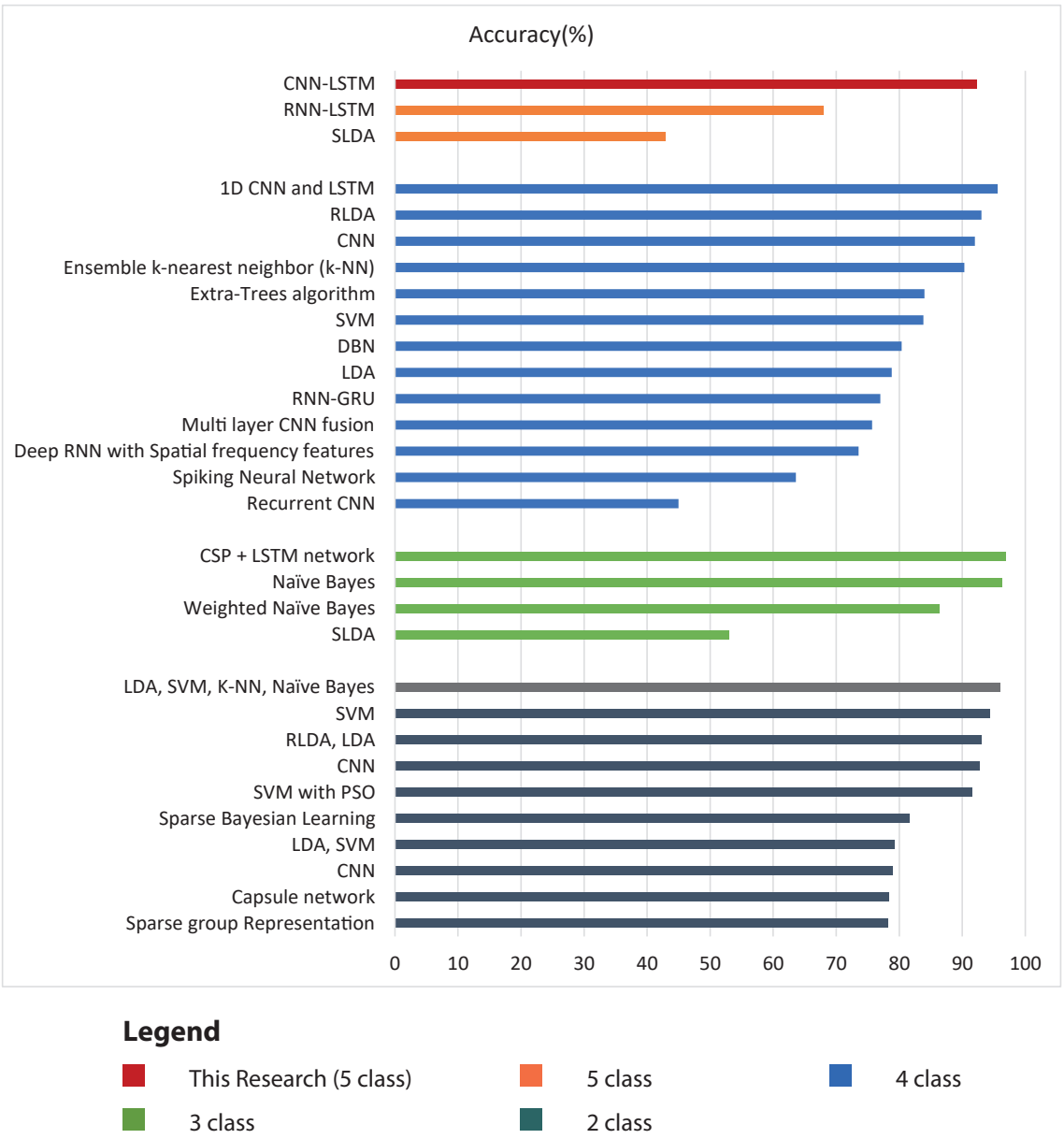


Figure 4.18: Comparison of classification accuracy across various related studies (refer Table 2.4).

4.3.8 Results with BCI IV 2a Dataset

To further validate the procedure of EEG classification and performance of the CNN-LSTM classifier architecture, a similar procedure of EEG classification is applied to BCI IV 2a dataset [27]. Training Data for a subject of BCI IVA 2a dataset is used to obtain classification performance as shown by Figure 4.19.

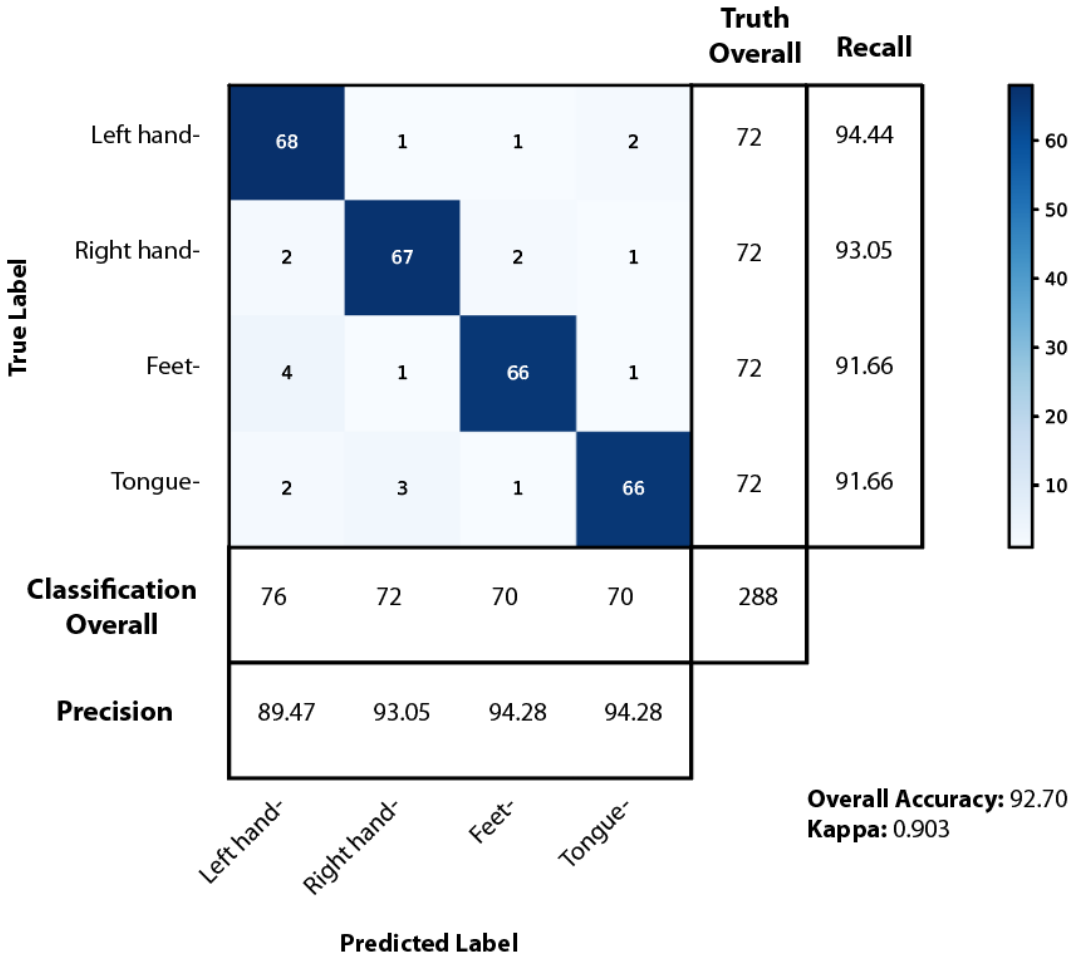


Figure 4.19: Confusion matrix for the training set of a subject of BCI IV 2a dataset.

This shows that, EEG classification approach proposed by this research shows similar classification performance on wide variety of datasets, patients/subjects, and tasks.

4.4 Chapter Summary

Good generalization with an average accuracy of 92.36% is achieved using hybrid CNN-LSTM network for the classification of attempted hand movements of SCI patients. A combination of FIR filter, downsampling, and ICA is used to remove unwanted components and increase the SNR of EEG data. TFR along with spatial encoding is used for the effective representation of EEG data for classification purposes. Sliding window is used with TFR data before feeding it into hybrid CNN-LSTM network and CNN-LSTM network architecture has been designed to:

- Learn spectral component of each channel.
- Learn spatial information of frequency components.
- Learn time series variation of localization information.
- Use time series information to classify 5 classes

This approach for classification of attempted movement of SCI patients has led to an increase of 47.36% classification accuracy compared to [20] for 5 class classification of attempted movement of SCI patients while 10% data for training.

CHAPTER 5
CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

Classification of attempted hand movement of SCI patients is a very important step in improving the quality of life of SCI patients. In this regard, this research has been able to accomplish the objective of developing and optimizing a hybrid CNN-LSTM model for multichannel EEG signal classification of attempted hand movement of SCI patients, and validating its classification performance.

5.1.1 Research Outcomes

Key aspects of this research can be concluded as follows:

(i) Classification accuracy of 92.36% with good generalization of the classifer is achieved using proposed method of EEG classification with hybrid CNN-LSTM network.

(ii) Calibration time of the EEG classifier is reduced by using 10% of the dataset for training purposes to achieve a classification accuracy of 92.36%.

(iii) Classification in done on EEG data from real-world SCI patients to classify five classes of single hand movements (pronation, supination, hand open, palmar grasp, and lateral grasp).

(iv) Proposed EEG classification procedure can extract and use spectral information, spatial information, and temporal information of valuable neural impulses from high dimensional EEG data.

(v) Artifact removal can be carried out using ICA to achieve maximum classification performance. Although, the step of artifact removal using ICA can be dropped to reduce computational complexity without much compromise on classification performance.

(vi) Proposed EEG classification method can be used to classify EEG data for other applications such as mental disease diagnosing, BCI applications, etc.

5.1.2 Practical Application

This research shows a method for achieving high classification performance for EEG signals. For the practical application of the proposed EEG classifier to classify hand movements of SCI patients, there are several things to be considered. High resolution acquisition of EEG signals for SCI patients is a challenging task to perform for regular use. There is a requirement for a plug-and-play wearable system for high resolution EEG imaging. Also, a fast communication system is required which is capable of transferring EEG data with low latency such that there is an acceptable lag between the attempted task and the transfer of EEG data to EEG classifier system. If the EEG data is acquired by the classification system within an acceptable delay, then we need to make a few adjustments to the proposed EEG classification system to work on real-time classification. EEG classification system should be made causal (e.g. FIR filters). Artifact removal using ICA is an effective but computationally intensive process. Thus, artifact removal using ICA should be avoided if the classification performance is sufficient enough for the patient because there is no significant decrease in classification performance without artifact removal using ICA. But, in some cases, artifact removal using ICA is a must to achieve expected classification performance. This depends on the nature of the patient and artifacts related to them. The computational time of post-processing and inference via hybrid CNN-LSTM network should be within an acceptable delay. For this purpose, a system with sufficient processing capabilities should be used.

On the actuator side, there is a requirement of robust, reliable, and ergonomic robotic hand system capable of performing hand movements. The robotic hand should be light weight and portable enough for daily use.

5.1.3 Limitations and Future Work

Inter subject performance With reference to the achieved classification performance of proposed EEG classification system, there is significantly low classification accuracy when the system is trained for one patient and tested for another patient. This is due to the fact that there are significant changes in EEG signal generated for the same task by two different patients. Further study is required to extract common EEG signals generated for a particular task irrespective of patients.

End to End DNN System An end-to-end DNN classifier can be constructed by further studies on the concept of CNN-LSTM architecture for EEG classification. Time-Frequency Representation of 61 EEG channels requires a lot of hardware requirement to store and load TFR data. A Convolutional Neural Network can be designed to perform the task of time-frequency representation on the CNN-LSTM network itself. This would reduce storage and computational requirement. Artifact removal using ICA is computationally intensive and there is a requirement of manually designing the encoding matrix according to the spatial information of EEG electrodes used. Artifact removal and spatial encoding can also be incorporated as part of DNN with further research in this field.

Number of classes Five classes of five different tasks are used in this research. It can further be increased to a higher number of classes for additional types of tasks.

Other applications The concepts of this research can be used not only for attempted movements of SCI patients but also for other neurological diseases, neuroscience applications, mental workload, neuromarketing, and biometrics in future works.

REFERENCES

[1] C. Guger, B. Z. Allison, and A. Gunduz, Eds., *Brain-Computer Interface Research: A State-of-the-Art Summary 10*, 1st ed. Springer, Aug. 2021.

[2] A. Kawala-Sterniuk *et al.*, "Summary of over Fifty Years with Brain-Computer Interfaces—A Review," *Brain Sciences*, vol. 11, no. 1, p. 43, Jan. 2021, number: 1 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/2076-3425/11/1/43

[3] A. Kumari and D. R. Edla, "A Study on Brain–Computer Interface: Methods and Applications," *SN Computer Science*, vol. 4, no. 2, p. 98, Dec. 2022. [Online]. Available: https://doi.org/10.1007/s42979-022-01515-0

[4] G. Muhammad, F. Alshehri, F. Karray, A. E. Saddik, M. Alsulaiman, and T. Falk, "A Comprehensive Survey on Multimodal Medical Signals Fusion for Smart Healthcare Systems," *Information Fusion*, vol. 76, pp. 355–375, Dec. 2021. [Online]. Available: https://dclibrary.mbzuai.ac.ae/mlfp/40

[5] S. Saha *et al.*, "Progress in Brain Computer Interface: Challenges and Opportunities," *Frontiers in Systems Neuroscience*, vol. 15, 2021. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnsys.2021.578875

[6] P. Olejniczak, "Neurophysiologic basis of EEG," *Journal of Clinical Neurophysiology: Official Publication of the American Electroencephalographic Society*, vol. 23, no. 3, pp. 186–189, Jun. 2006.

[7] D. L. Schomer and F. H. Lopes da Silva, Eds., *Niedermeyer's Electroencephalography: Basic Principles, Clinical Applications, and Related Fields*. Oxford University Press, Nov. 2017. [Online]. Available: https://doi.org/10.1093/med/9780190228484.001.0001

[8] R. P. Turner, "Clinical Application of Combined EEG-qEEG Functional Neuroimaging in the Practice of Pediatric Neuroscience: A Personal Perspective," *Clinical EEG and Neuroscience*, vol. 52, no. 2, pp. 126–135, Apr. 2021, publisher: SAGE Publications Inc. [Online]. Available: https://doi.org/10.1177/1550059420982419

[9] M. Teplan, "Fundamental of EEG Measurement," *MEASUREMENT SCIENCE REVIEW*, vol. 2, Jan. 2002.

[10] F. Lotte, "A Tutorial on EEG Signal-processing Techniques for Mental-state Recognition in Brain–Computer Interfaces," in *Guide to Brain-Computer Music Interfacing*, E. R. Miranda and J. Castet, Eds. London: Springer, 2014, pp. 133–161. [Online]. Available: https://doi.org/10.1007/978-1-4471-6584-2_7

[11] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, and K.-r. Muller, "Optimizing Spatial filters for Robust EEG Single-Trial Analysis," *IEEE Signal Processing Magazine*, vol. 25, no. 1, pp. 41–56, 2008, conference Name: IEEE Signal Processing Magazine.

[12] M. Clerc, L. Bougrain, and F. Lotte, *Brain-computer interfaces 2: technology and applications*. John Wiley & Sons, 2016.

[13] F. Lotte *et al.*, "A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update," *Journal of Neural Engineering*, vol. 15, no. 3, p. 031005, Apr. 2018, publisher: IOP Publishing. [Online]. Available: https://dx.doi.org/10.1088/1741-2552/aab2f2

[14] B. Graimann, B. Z. Allison, and G. Pfurtscheller, *Brain-computer interfaces: Revolutionizing human-computer interaction*. Springer Science & Business Media, 2010.

[15] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain–computer interfaces for communication and control," *Clinical Neurophysiology*, vol. 113, no. 6, pp. 767–791, Jun. 2002. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1388245702000573

[16] D. J. Krusienski *et al.*, "Critical issues in state-of-the-art brain–computer interface signal processing," *Journal of Neural Engineering*, vol. 8, no. 2, p. 025002, Mar. 2011. [Online]. Available: https://dx.doi.org/10.1088/1741-2560/8/2/025002

[17] F. Lotte and C. Jeunet, "Online classification accuracy is a poor metric to study mental imagery-based bci user learning: an experimental demonstration and new metrics," in *Online classification accuracy is a poor metric to study mental imagery-based bci user learning: an experimental demonstration and new metrics*, Sep. 2017. [Online]. Available: https://hal.science/hal-01519478

[18] J. Millan, "On the need for on-line learning in brain-computer interfaces," in *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541)*, vol. 4, Jul. 2004, pp. 2877–2882 vol.4, iSSN: 1098-7576.

[19] P. Shenoy, M. Krauledat, B. Blankertz, R. P. N. Rao, and K.-R. Müller, "Towards adaptive classification for BCI*," *Journal of Neural Engineering*, vol. 3, no. 1, p. R13, Mar. 2006. [Online]. Available: https://dx.doi.org/10.1088/1741-2560/3/1/R02

[20] P. Ofner, A. Schwarz, J. Pereira, D. Wyss, R. Wildburger, and G. R. Müller-Putz, "Attempted Arm and Hand Movements can be Decoded from Low-Frequency EEG from Persons with Spinal Cord Injury," *Scientific Reports*, vol. 9, no. 1, p. 7134, May 2019. [Online]. Available: https://doi.org/10.1038/s41598-019-43594-9

[21] A. F. Jackson and D. J. Bolger, "The neurophysiological bases of EEG and EEG measurement: A review for the rest of us," *Psychophysiology*, vol. 51, no. 11, pp. 1061–1071, 2014, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/psyp.12283. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/psyp.12283

[22] C. Babiloni *et al.*, "International Federation of Clinical Neurophysiology (IFCN) – EEG research workgroup: Recommendations on frequency and topographic analysis of resting state EEG rhythms. Part 1: Applications in

clinical research studies," *Clinical Neurophysiology*, vol. 131, no. 1, pp. 285–307, Jan. 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1388245719311642

[23] M.-P. Hosseini, A. Hosseini, and K. Ahi, "A Review on Machine Learning for EEG Signal Processing in Bioengineering," *IEEE Reviews in Biomedical Engineering*, vol. 14, pp. 204–218, 2021, conference Name: IEEE Reviews in Biomedical Engineering.

[24] S. Aggarwal and N. Chugh, "Review of Machine Learning Techniques for EEG Based Brain Computer Interface," *Archives of Computational Methods in Engineering*, vol. 29, no. 1, pp. 1–20, Jan. 2022. [Online]. Available: https://doi.org/10.1007/s11831-021-09684-6

[25] 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, conference Name: Proceedings of the IEEE.

[26] 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. L14, Aug. 2005. [Online]. Available: https://dx.doi.org/10.1088/1741-2560/2/4/L02

[27] M. Tangermann *et al.*, "Review of the BCI Competition IV," *Frontiers in Neuroscience*, vol. 6, p. 55, 2012.

[28] R. Oostenveld and P. Praamstra, "The five percent electrode system for high-resolution EEG and ERP measurements," *Clinical Neurophysiology: Official Journal of the International Federation of Clinical Neurophysiology*, vol. 112, no. 4, pp. 713–719, Apr. 2001.

[29] A. Delorme, T. Sejnowski, and S. Makeig, "Enhanced detection of artifacts in EEG data using higher-order statistics and independent component analysis," *NeuroImage*, vol. 34, no. 4, pp. 1443–1449, Feb. 2007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1053811906011098

[30] A. Kilicarslan, R. G. Grossman, and J. L. Contreras-Vidal, "A robust adaptive denoising framework for real-time artifact removal in scalp EEG measurements," *Journal of Neural Engineering*, vol. 13, no. 2, p. 026013, Feb. 2016, publisher: IOP Publishing. [Online]. Available: https://dx.doi.org/10.1088/1741-2560/13/2/026013

[31] J. E. Kline, H. J. Huang, K. L. Snyder, and D. P. Ferris, "Isolating gait-related movement artifacts in electroencephalography during human walking," *Journal of Neural Engineering*, vol. 12, no. 4, p. 046022, Jun. 2015, publisher: IOP Publishing. [Online]. Available: https://dx.doi.org/10.1088/1741-2560/12/4/046022

[32] J. X. Chen, Z. J. Mao, W. X. Yao, and Y. F. Huang, "EEG-based biometric identification with convolutional neural network," *Multimedia Tools and Applications*, vol. 79, no. 15-16, pp. 10 655–10 675, Apr. 2020. [Online]. Available: https://doi.org/10.1007/s11042-019-7258-4

[33] P. Ofner and G. R. Muller-Putz, "Movement target decoding from EEG and the corresponding discriminative sources: A preliminary study," *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference*, vol. 2015, pp. 1468–1471, Aug. 2015.

[34] P. Ofner and G. R. Müller-Putz, "Using a noninvasive decoding method to classify rhythmic movement imaginations of the arm in two planes," *IEEE transactions on bio-medical engineering*, vol. 62, no. 3, pp. 972–981, Mar. 2015.

[35] X. Jiang, G.-B. Bian, and Z. Tian, "Removal of Artifacts from EEG Signals: A Review," *Sensors (Basel, Switzerland)*, vol. 19, no. 5, p. 987, Feb. 2019.

[36] L. Sörnmo and P. Laguna, "Chapter 7 - ECG Signal Processing," in *Bioelectrical Signal Processing in Cardiac and Neurological Applications*, ser. Biomedical Engineering, L. Sörnmo and P. Laguna, Eds. Burlington: Academic Press, Jan. 2005, pp. 453–566. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780124375529500076

[37] A. Craik, Y. He, and J. L. Contreras-Vidal, "Deep learning for electroencephalogram (EEG) classification tasks: a review," *Journal of Neural Engineering*, vol. 16, no. 3, p. 031001, Apr. 2019, publisher: IOP Publishing. [Online]. Available: https://dx.doi.org/10.1088/1741-2552/ab0ab5

[38] A. Khosla, P. Khandnor, and T. Chand, "A comparative analysis of signal processing and classification methods for different applications based on EEG signals," *Biocybernetics and Biomedical Engineering*, vol. 40, no. 2, pp. 649–690, Apr. 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0208521620300231

[39] J. Pereira, P. Ofner, and G. R. Müller-Putz, "Goal-directed or aimless? EEG differences during the preparation of a reach-and-touch task," in *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Aug. 2015, pp. 1488–1491, iSSN: 1558-4615.

[40] R. Chatterjee, A. Datta, and D. K. Sanyal, "Chapter 8 - Ensemble Learning Approach to Motor Imagery EEG Signal Classification," in *Machine Learning in Bio-Signal Analysis and Diagnostic Imaging*, N. Dey, S. Borra, A. S. Ashour, and F. Shi, Eds. Academic Press, Jan. 2019, pp. 183–208. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780128160862000084

[41] P. Ofner, A. Schwarz, J. Pereira, and G. Müller-Putz, "Movements of the same upper limb can be classified from low-frequency time-domain EEG signals: 6th International BCI Meeting," *Proceedings of the Sixth International Brain-Computer Interface Meeting: BCI Past, Present, and Future*, p. 69, 2016, publisher: Verlag der Technischen Universität Graz.

[42] J. Zhang and S. Li, "A deep learning scheme for mental workload classification based on restricted Boltzmann machines," *Cognition, Technology & Work*, vol. 19, no. 4, pp. 607–631, Nov. 2017. [Online]. Available: https://doi.org/10.1007/s10111-017-0430-6

[43] 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, Jan. 2019, number: 6 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/1424-8220/19/6/1423

[44] 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, Jan. 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1746809416301331

[45] A. Graps, "An introduction to wavelets," *IEEE Computational Science and Engineering*, vol. 2, no. 2, pp. 50–61, 1995, conference Name: IEEE Computational Science and Engineering.

[46] Y. Kutlu and D. Kuntalp, "Feature extraction for ECG heartbeats using higher order statistics of WPD coefficients," *Computer Methods and Programs in Biomedicine*, vol. 105, no. 3, pp. 257–267, Mar. 2012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169260711002665

[47] M. Li, X. Luo, J. Yang, and Y. Sun, "Applying a Locally Linear Embedding Algorithm for Feature Extraction and Visualization of MI-EEG," *Journal of Sensors*, vol. 2016, p. e7481946, Aug. 2016, publisher: Hindawi. [Online]. Available: https://www.hindawi.com/journals/js/2016/7481946/

[48] S. Kumar, A. Sharma, and T. Tsunoda, "An improved discriminative filter bank selection approach for motor imagery EEG signal classification using mutual information," *BMC Bioinformatics*, vol. 18, no. 16, p. 545, Dec. 2017. [Online]. Available: https://doi.org/10.1186/s12859-017-1964-6

[49] S. Kumar, R. Sharma, A. Sharma, and T. Tsunoda, "Decimation filter with Common Spatial Pattern and Fishers Discriminant Analysis for motor imagery classification," in *2016 International Joint Conference on Neural Networks (IJCNN)*, Jul. 2016, pp. 2090–2095, iSSN: 2161-4407.

[50] L. F. Velásquez-Martínez, A. M. Álvarez Meza, and C. G. Castellanos-Domínguez, "Motor Imagery Classification for BCI Using Common Spatial Patterns and Feature Relevance Analysis," in *Natural and Artificial Computation in Engineering and Medical Applications*, ser. Lecture Notes in Computer Science, J. M. Ferrández Vicente, J. R. Álvarez Sánchez, F. de la Paz López, and F. J. Toledo Moreo, Eds. Berlin, Heidelberg: Springer, 2013, pp. 365–374.

[51] X. Yu, P. Chum, and K.-B. Sim, "Analysis the effect of PCA for feature reduction in non-stationary EEG based motor imagery of BCI system," *Optik*, vol. 125, pp. 1498–1502, Feb. 2014.

[52] L. Moctezuma, A. Torres-García, L. Villaseñor-Pineda, and M. Carrillo, "Subjects Identification using EEG-recorded Imagined Speech," *Expert Systems with Applications*, vol. 118, pp. 201–208, Mar. 2019.

[53] D. J. McFarland, "The advantages of the surface Laplacian in brain–computer interface research," *International Journal of Psychophysiology*, vol. 97, no. 3, pp. 271–276, Sep. 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S016787601400186X

[54] R. Chatterjee, T. Maitra, S. Hafizul Islam, M. M. Hassan, A. Alamri, and G. Fortino, "A novel machine learning based feature selection for motor imagery EEG signal classification in Internet of medical things environment," *Future Generation Computer Systems*, vol. 98, pp. 419–434, Sep. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X18327699

[55] 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)*, Feb. 2018, pp. 1–5.

[56] G. Korats, S. Le Cam, R. Ranta, and M. Hamid, "Applying ICA in EEG: Choice of the Window Length and of the Decorrelation Method," in *Biomedical Engineering Systems and Technologies*, ser. Communications in Computer and Information Science, J. Gabriel *et al.*, Eds. Berlin, Heidelberg: Springer, 2013, pp. 269–286.

[57] T. Liu and D. Yao, "Removal of the ocular artifacts from EEG data using a cascaded spatio-temporal processing," *Computer Methods and Programs in Biomedicine*, vol. 83, no. 2, pp. 95–103, Aug. 2006.

[58] D. T. Pham and P. Garat, "Blind separation of mixture of independent sources through a quasi-maximum likelihood approach," *IEEE Transactions on Signal Processing*, vol. 45, pp. 1712–1725, Jul. 1997, aDS Bibcode: 1997ITSP...45.1712P. [Online]. Available: https://ui.adsabs.harvard.edu/abs/1997ITSP...45.1712P

[59] P. Ablin, J.-F. Cardoso, and A. Gramfort, "Faster Independent Component Analysis by Preconditioning With Hessian Approximations," *IEEE Transactions on Signal Processing*, vol. 66, no. 15, pp. 4040–4049, Aug. 2018, conference Name: IEEE Transactions on Signal Processing.

[60] P. Comon, "Independent component analysis, A new concept?" *Signal Processing*, vol. 36, no. 3, pp. 287–314, Apr. 1994. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0165168494900299

[61] A. Gramfort *et al.*, "MEG and EEG Data Analysis with MNE-Python," *Frontiers in Neuroscience*, vol. 7, no. 267, pp. 1–13, 2013.

[62] Y. Xie and S. Oniga, "A Review of Processing Methods and Classification Algorithm for EEG Signal," *Carpathian Journal of Electronic and Computer Engineering*, vol. 13, no. 1, pp. 23–29, Sep. 2020. [Online]. Available: https://www.sciendo.com/article/10.2478/cjece-2020-0004

[63] Q. Huang, Z. Zhang, T. Yu, S. He, and Y. Li, "An EEG-/EOG-Based Hybrid Brain-Computer Interface: Application on Controlling an Integrated Wheelchair Robotic Arm System," *Frontiers in Neuroscience*, vol. 13, 2019. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnins.2019.01243

[64] W. W. Sakti, K. Anam, S. B. Utomo, B. Marhaenanto, and S. Nahela, "Artificial Intelligence IoT based EEG Application using Deep Learning for Movement Classification," in *2021 8th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, Oct. 2021, pp. 192–196.

[65] W. Zgallai *et al.*, "Deep Learning AI Application to an EEG driven BCI Smart Wheelchair," in *2019 Advances in Science and Engineering Technology International Conferences (ASET)*, Mar. 2019, pp. 1–5.

[66] 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, Nov. 2018, conference Name: IEEE Transactions on Neural Systems and Rehabilitation Engineering.

[67] Y. Jiao *et al.*, "Sparse Group Representation Model for Motor Imagery EEG Classification," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 2, pp. 631–641, Mar. 2019, conference Name: IEEE Journal of Biomedical and Health Informatics.

[68] K.-W. Ha and J.-W. Jeong, "Motor Imagery EEG Classification Using Capsule Networks," *Sensors*, vol. 19, no. 13, p. 2854, Jan. 2019, number: 13 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/1424-8220/19/13/2854

[69] H. K. Lee and Y.-S. Choi, "A convolution neural networks scheme for classification of motor imagery EEG based on wavelet time-frequecy image," in *2018 International Conference on Information Networking (ICOIN)*, Jan. 2018, pp. 906–909.

[70] 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)*, Jun. 2017, pp. 781–786, iSSN: 2372-9198.

[71] Y. Zhang, Y. Wang, J. Jin, and X. Wang, "Sparse Bayesian Learning for Obtaining Sparsity of EEG Frequency Bands Based Feature Vectors in Motor Imagery Classification," *International Journal of Neural Systems*, vol. 27, no. 02, p. 1650032, Mar. 2017, publisher: World Scientific Publishing Co. [Online]. Available: https://www.worldscientific.com/doi/abs/10.1142/S0129065716500325

[72] Y. Ma, X. Ding, Q. She, Z. Luo, T. Potter, and Y. Zhang, "Classification of Motor Imagery EEG Signals with Support Vector Machines and Particle Swarm Optimization," *Computational and Mathematical Methods in Medicine*, vol. 2016, p. 4941235, 2016.

[73] R. Fu, Y. Tian, T. Bao, Z. Meng, and P. Shi, "Improvement Motor Imagery EEG Classification Based on Regularized Linear Discriminant Analysis," *Journal of Medical Systems*, vol. 43, no. 6, p. 169, May 2019. [Online]. Available: https://doi.org/10.1007/s10916-019-1270-0

[74] M. Z. Baig, N. Aslam, H. P. Shum, and L. Zhang, "Differential evolution algorithm as a tool for optimal feature subset selection in motor imagery EEG," *Expert Systems With Applications*, vol. 90, pp. 184–195, Dec. 2017. [Online]. Available: http://www.scopus.com/inward/record.url?scp=85027505525&partnerID=8YFLogxK

[75] M. Miao, H. Zeng, A. Wang, C. Zhao, and F. Liu, "Discriminative spatial-frequency-temporal feature extraction and classification of motor imagery EEG: An sparse regression and Weighted Naïve Bayesian Classifier-based approach," *Journal of Neuroscience Methods*, vol. 278, pp. 13–24, Feb. 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0165027016302990

[76] Siuly, H. Wang, and Y. Zhang, "Detection of motor imagery EEG signals employing Naïve Bayes based learning process," *Measurement*, vol. 86, pp. 148–158, May 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0263224116001469

[77] F. M. Garcia-Moreno, M. Bermudez-Edo, M. J. Rodríguez-Fórtiz, and J. L. Garrido, "A CNN-LSTM Deep Learning Classifier for Motor Imagery EEG Detection Using a Low-invasive and Low-Cost BCI Headband," in *2020 16th International Conference on Intelligent Environments (IE)*, Jul. 2020, pp. 84–91, iSSN: 2472-7571.

[78] S. Kumar, A. Sharma, and T. Tsunoda, "Brain wave classification using long short-term memory network based OPTICAL predictor," *Scientific Reports*, vol. 9, no. 1, p. 9153, Jun. 2019, number: 1 Publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41598-019-45605-1

[79] W. Ko, J. Yoon, E. Kang, E. Jun, J.-S. Choi, and H.-I. Suk, "Deep recurrent spatio-temporal neural network for motor imagery based BCI," in *2018 6th International Conference on Brain-Computer Interface (BCI)*, Jan. 2018, pp. 1–3, iSSN: 2572-7672.

[80] W. Abbas and N. A. Khan, "DeepMI: Deep Learning for Multiclass Motor Imagery Classification," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Jul. 2018, pp. 219–222, iSSN: 1558-4615.

[81] R. Salazar-Varas and R. A. Vazquez, "Evaluating spiking neural models in the classification of motor imagery EEG signals using short calibration sessions," *Applied Soft Computing*, vol. 67, pp. 232–244, Jun. 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1568494618301133

[82] S. Sakhavi, C. Guan, and S. Yan, "Parallel convolutional-linear neural network for motor imagery classification," in *2015 23rd European Signal Processing Conference (EUSIPCO)*, Aug. 2015, pp. 2736–2740, iSSN: 2076-1465.

[83] T.-j. Luo, C.-l. Zhou, and F. Chao, "Exploring spatial-frequency-sequential relationships for motor imagery classification with recurrent neural network," *BMC Bioinformatics*, vol. 19, no. 1, p. 344, Sep. 2018. [Online]. Available: https://doi.org/10.1186/s12859-018-2365-1

[84] 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, Nov. 2018, conference Name: IEEE Transactions on Neural Networks and Learning Systems.

[85] S. U. Amin, M. Alsulaiman, G. Muhammad, M. A. Mekhtiche, and M. Shamim Hossain, "Deep Learning for EEG motor imagery classification based on multi-layer CNNs feature fusion," *Future Generation Computer Systems*, vol. 101, pp. 542–554, Dec. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X19306077

[86] P. Lu, N. Gao, Z. Lu, J. Yang, O. Bai, and Q. Li, "Combined CNN and LSTM for Motor Imagery Classification," in *2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, Oct. 2019, pp. 1–6.

[87] R. Aler, I. M. Galvan, and J. M. Valls, "Evolving spatial and frequency selection filters for Brain-Computer Interfaces," in *IEEE Congress on Evolutionary Computation*. Barcelona, Spain: IEEE, Jul. 2010, pp. 1–7. [Online]. Available: http://ieeexplore.ieee.org/document/5586383/

[88] L. Q. Thang and C. Temiyasathit, "Increase performance of four-class classification for motor-imagery based brain-computer interface," in *2014 International Conference on Computer, Information and Telecommunication Systems (CITS)*, Jul. 2014, pp. 1–5.

[89] Z. Yu and J. Song, "Multi-class motor imagery classification by singular value decomposition and deep boltzmann machine," in *2017 IEEE 3rd Information Technology and Mechatronics Engineering Conference (ITOEC)*, Oct. 2017, pp. 376–379.

[90] M. R. Islam, T. Tanaka, M. S. Akter, and M. K. I. Molla, "Classification of motor imagery BCI using multiband tangent space mapping," in *2017 22nd International Conference on Digital Signal Processing (DSP)*, Aug. 2017, pp. 1–5, iSSN: 2165-3577.

[91] S. Bera, R. Roy, D. Sikdar, A. Kar, R. Mukhopadhyay, and M. Mahadevappal, "A Randomised Ensemble Learning Approach for Multiclass Motor Imagery Classification Using Error Correcting Output Coding," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Jul. 2018, pp. 5081–5084, iSSN: 1558-4615.

[92] M. Rashid *et al.*, "The classification of motor imagery response: an accuracy enhancement through the ensemble of random subspace k-NN," *PeerJ Computer Science*, vol. 7, p. e374, Mar. 2021. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7959631/

[93] Z. Khademi, F. Ebrahimi, and H. M. Kordy, "A transfer learning-based CNN and LSTM hybrid deep learning model to classify motor imagery EEG signals," *Computers in Biology and Medicine*, vol. 143, p. 105288, Feb. 2022.

[94] C. Uyulan, "Development of LSTM&CNN based hybrid deep learning model to classify motor imagery tasks," *Communications in Mathematical Biology and Neuroscience*, vol. 2021, Jan. 2021.

[95] L. Jingwei, C. Yin, and Z. Weidong, "Deep learning EEG response representation for brain computer interface," in *2015 34th Chinese Control Conference (CCC)*, Jul. 2015, pp. 3518–3523, iSSN: 1934-1768.

[96] X. Ma, S. Qiu, C. Du, J. Xing, and H. He, "Improving EEG-Based Motor Imagery Classification via Spatial and Temporal Recurrent Neural Networks," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Jul. 2018, pp. 1903–1906, iSSN: 1558-4615.

[97] E. A. Aydin, "Classification of Forearm Movements by Using Movement Related Cortical Potentials," in *2022 Innovations in Intelligent Systems and Applications Conference (ASYU)*, Sep. 2022, pp. 1–4, iSSN: 2770-7946.

[98] Martín Abadi *et al.*, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems," 2015. [Online]. Available: https://www.tensorflow.org/

[99] D. Dharmaprani, H. K. Nguyen, T. W. Lewis, D. DeLosAngeles, J. O. Willoughby, and K. J. Pope, "A comparison of independent component analysis algorithms and measures to discriminate between EEG and artifact components," in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Aug. 2016, pp. 825–828, iSSN: 1558-4615.

[100] R. Kronland-Martinet, "The Wavelet Transform for Analysis, Synthesis, and Processing of Speech and Music Sounds," *Computer Music Journal*, vol. 12, Dec. 1988.

[101] M. X. Cohen, "A better way to define and describe Morlet wavelets for time-frequency analysis," *NeuroImage*, vol. 199, pp. 81–86, Oct. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1053811919304409

[102] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," Mar. 2015, arXiv:1502.03167 [cs]. [Online]. Available: http://arxiv.org/abs/1502.03167

[103] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-k. Wong, and W.-c. Woo, "Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting," Jun. 2015, arXiv:1506.04214 [cs] version: 1. [Online]. Available: http://arxiv.org/abs/1506.04214

[104] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," Jan. 2017, arXiv:1412.6980 [cs]. [Online]. Available: http://arxiv.org/abs/1412.6980

[105] M. L. McHugh, "Interrater reliability: the kappa statistic," *Biochemia Medica*, vol. 22, no. 3, pp. 276–282, Oct. 2012. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3900052/

APPENDIX A
PICARD Method of ICA

PICARD stands for Preconditioned Independent Component Analysis of Real Data [59]. For large set of real data, the maximization of the likelihood is a demanding issue. Even more so, if it has to be completed quickly and accurately. One way of solving this problem is to use quasi-Newton methods which rely upon sparse approximations of the Hessian of the log-likelihood, but doesnot hold for real datasets where sources are not completely independent. PICARD is a method which makes use of sparse approximatation of Hessian only as a preconditioner to the L-BFGS algorithm. Hessian approximation is refined from a memory of the past iterates [59]. Rewriting log likelihood of probability density function as:

$$L(W) = -log|det\ (W)| - \hat{E}\ \left[\sum_{i=1}^{N} log(p_i(y_i(t)))\right] \tag{A.1}$$

Where $\hat{E}$ denotes the empirical mean (sample average), W is mixing matrix, p_i is probability density function, $y_i(t)$ is source values at time t.

The variation of $L(W)$ with respect to a relative variation of W upto second order as described by Taylor Expansion of $L((I+E)W)$ in terms of 'small' NxN matrix E, along with first order G_{ij} and second order Hessian H_{ijkl}:

$$L((I+E)W) = L(W) + < G|E > + \frac{1}{2} < E|H|E > \tag{A.2}$$

$$G_{ij} = \hat{E}\ [\psi_i(y_i)y_j] - \delta_{ij}\ \ or\ \ G(Y) = \frac{1}{T}\psi(Y)Y^T - Id \tag{A.3}$$

$$H_{ijkl} = \delta_{il}\delta_{jk} + \delta_{ik}\ \hat{E}[\psi'(y_i)y_j y_l] \tag{A.4}$$

Hessian Approximations The Hessian approximations and regularization can be donoted as follows [59]:

$$\tilde{H}^2_{ijkl} = \delta_{il}\delta_{jk} + \delta_{ik}\delta_{jl}\ \hat{h}_{jk} \tag{A.5}$$

PICARD algortihm **Input**: Mixed signals X, initial unmixing matrix W_0

Set $Y = W_0 X$

For $k = 0, 1, \ldots, k$ do

Compute relative gradient G_k

Computer Hessian approximation $\tilde{H}_k$

Regularize $\tilde{H}_k$

Compute the search direction $p_k = -(\tilde{H}_k)^{-1} G_k$

Set $W_{k+1} = (I + \alpha_k p_k) W_k$ (α_k set by line search)

Set $Y = (I + \alpha_k p_k) Y$

End

Output: Y, W_k