EEG Channel Selection using Evolutionary Algorithm for BCI application

A Thesis Submitted
In Partial Fulfillment of the Requirements for the
Degree of

MASTER OF TECHNOLOGY
in
SIGNAL PROCESSING AND DIGITAL DESIGN

by

Ashish Karaiya
(2K23/SPD/09)

Under the supervision of
Dr. Sachin Taran

Department of Electronics and Communication Engineering

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-110042, India

MAY 2025

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-42

CANDIDATE'S DECLARATION

I, **Ashish Karaiya**, Roll No **2K23/SPD/09**, hereby certify that the work which is being presented in the thesis entitled "**EEG Channel Selection using Evolutionary Algorithm for BCI application**" in partial fulfillment of the requirements for the award of Degree of **Master of Technology**, submitted in the Department of **Electronics and Communication Engineering**, Delhi Technological University is an authentic record of my own work carried out during the period from **Dec 2024** to **May 2025** under the supervision of **Dr. Sachin Taran**. The matter presented in the thesis has not been submitted by me for the award of any other degree of this or any other Institute.

Candidate's Signature

<u>CERTIFICATE</u>

I hereby certify that the project dissertation titled "**EEG Channel Selection using Evolutionary Algorithm for BCI application**", which is submitted by **Ashish Karaiya**, Roll No. **2K23/SPD/09**, Department of Electronics and Communication Engineering, Delhi Technological University, Delhi in partial fulfilment of the requirements for the award of degree of Master of Technology in **Signal Processing and Digital Design** is a record of the project work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this university or elsewhere.

Place: Delhi
Date: 31st may 2025

Signature

Dr. Sachin Taran

Assistant Professor, Dept. of ECE
DTU, Shahbad Daulatpur,
Main Bawana Road, Delhi-42

ABSTRACT

Brain-Computer Interfaces (BCIs) offer a revolutionary means of communication between the human brain and external systems, enabling individuals to control devices without the need for any muscular activity. This technology holds immense potential in various domains, particularly in neurorehabilitation, where it assists patients with motor impairments, and in assistive technologies, providing control over prosthetic limbs, wheelchairs, and communication devices. Among the various BCI paradigms, Motor Imagery (MI) has emerged as one of the most promising approaches. In MI-based BCIs, users are trained to imagine specific movements—such as moving the left or right hand—without performing any actual physical motion. To monitor brain activity, Electroencephalography (EEG) is the most widely used modality in MI-based BCIs. EEG captures electrical activity from the scalp with high temporal resolution, making it particularly effective for tracking the fast neural dynamics associated with motor imagery. Additionally, EEG is non-invasive, portable, and cost-effective, offering significant advantages over other neuroimaging techniques such as fMRI and MEG. However, EEG signals are inherently high-dimensional and noise-prone, with artifacts stemming from muscle movement, eye blinks, and external interference. These challenges necessitate robust feature extraction and channel selection techniques to ensure accurate and efficient classification of MI tasks. Identifying the most informative channels and transforming the raw EEG into meaningful features are critical steps in reducing redundancy, improving signal quality, and enabling reliable real-time performance.

In this work, we propose an enhanced EEG-based framework for binary motor imagery classification, focusing specifically on distinguishing left- vs. right-hand imagery. To extract discriminative features, we employ Advanced Graph Signal Processing (AGSP), a novel approach that treats EEG signals as data on a graph, where nodes represent EEG channels and edges capture functional connectivity between brain regions. This graph-based representation enables the extraction of features that incorporate both spatial and structural information, offering deeper insights into brain dynamics compared to traditional time-series analysis. AGSP leverages graph spectral transforms to highlight connectivity-driven neural patterns relevant to MI classification. To further improve system performance, we implement the Set-based Integer-coded Fuzzy Granular Evolutionary (SIFE) algorithm for intelligent channel selection. SIFE utilizes swarm intelligence to explore the search space and select the most informative subset of EEG channels, effectively reducing dimensionality while preserving key discriminative features. This not only boosts classification accuracy but also reduces computational cost and enhances the system's real-time capabilities. For the final decision-making, we adopt an ensemble classification strategy by integrating multiple classifiers such as Random Forest, XG-Boost, and Ada-Boost. Ensemble learning enhances robustness and generalization by combining the strengths of individual models. The integration of AGSP for structural feature extraction, SIFE for optimized channel selection, and ensemble learning for robust classification results in a highly efficient and accurate framework for decoding binary motor imagery tasks from EEG signals.

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-42

ACKNOWLEDGEMENT

I, **Ashish Karaiya**, Roll No. **2K23/SPD/09**, student of **M.Tech (Signal Processing and Digital Design**), hereby thank and express my sincere gratitude to my supervisor, **Dr. Sachin Taran**, with whose continuous support and insightful guidance, this project titled "**EEG Channel Selection using Evolutionary Algorithm for BCI application**" was successfully undertaken by me.

Place: Delhi   **Ashish Karaiya**

Date: 31st may 2025   **(2K23/SPD/09)**

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

ADA	Adaptive Boosting
AGSP	Advanced graph signal processing
BCI	Brain Computer Interface
CNN	Convolutional Neural Networks
CPU	Central Processing Unit
DWT	Discrete Wavelet Transform
GPU	Graphics Processing Unit
GSP	Graph Signal Processing
GFT	Graph Fourier Transform
EEG	Electroencephalogram
FN	False Negative
FBCSP	Filter Bank Common Spatial Pattern
FP	False Positive
NumPy	Numerical Python
MI	Motor Imagery
Pandas	Python Data Analysis Library
PSO	Particle swarm optimization
RF	Random Forest
ROC	Receiver Operating Characteristic
RMS	Root Mean Square
SIFE	Set-based Integer-coded Fuzzy granular Evolutionary
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
TPU	Tensor Processing Unit
XGB	Extreme Gradient Boosting

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1. OVERVIEW:

Brain-computer interfaces (BCI) create a straight-forward communication channel between the brain and external equipment bypassing traditional output routes like the muscles or speech. BCIs are essential in many areas, especially in the areas of assistive technology and rehabilitation for people with motor impairments. Motor Imagery (MI) is a fundamental approach in BCIs [1]. The ability to mentally simulate movement without actually performing it physically is known as motor imagery. When a person imagines moving, their brain generates electrical patterns similar to those produced during actual motor execution. This makes MI a useful tool in BCIs, as it enables users to operate devices or communicate through the imagination of movement. This approach has found significant application in medical fields, especially in patients recovering from stroke, spinal cord injury, or other motor impairing illnesses [2].

1.1.1. Motor Imagery in BCI:
MI can be [3]used to identify certain brain activity patterns and translate them into commands for operating external devices like computer cursors, robotic limbs, or exoskeletons. It has also demonstrated potential in controlling assistive technologies for severely disabled people, holding new promise for independence and interaction with environment [4].

1.1.2. Inputs for Motor Imagery Detection:
Electroencephalography (EEG), which analyses the electrical activity of the brain, is the main input utilized to identify MI signals. EEG records brain activity in real time without intrusive procedures, capturing the neural impulses linked to motor imagery. These signals are then converted into control signals for BCI applications after being processed, examined, and evaluated [5].

1.1.3. Why EEG Dominates Other Modalities in BCI:
However, there are additional methods, including near-infrared spectroscopy (NIRS), functional magnetic resonance imaging (fMRI [6]), MEG [7], and ECoG for identifying brain activity [8]. In BCI applications,

EEG continues to be the most used modality. This is because of several factors:

- **Non-invasive and Cost-effective:** EEG does not require surgery or the use of expensive equipment, making it a more practical and affordable option.

- **High Temporal Resolution:** EEG provides real-time data, allowing for immediate feedback and quick response times, which is crucial in BCI applications, especially for motor imagery.

- **Portability:** EEG systems are generally lightweight and can be easily worn by users in various **environments**, which is particularly useful for both clinical and everyday BCI applications.

Because of these benefits, EEG is the preferred method for tracking brain activity in MI-based BCIs, enabling the creation of accessible and efficient assistive devices.

1.1.4. Electroencephalography:

Brain activity generates a variety of signals, including electrical and magnetic impulses. This activity can be recorded in a variety of ways, which are often divided into non-invasive and invasive categories. Invasive approaches contain surgery to put a certain device in the brain, whereas non-invasive approaches do not contain any surgery. Electroencephalography [9, 10] is most frequently used non-invasive ways to capture brain signals. EEG is a basic and easy way to capture brain electrical activity without having to cut into the brain. The activity is shown as voltage changes that happen when current flows through the neurons in the brain. The electrodes on the scalp record EEG waves, which can be thought of as the signal over time. EEG is a way to measure voltage over time. The activity level of the cerebral cortex has a big effect on EEG characteristics. The EEG signal is made up of a variety of waveforms, and it is usually grouped by its 1) frequency 2) Size 3) Shape of the wave 4) Distribution in space 5) Reactivity. The most popular way to categorize EEG signals is by the frequency band of the EEG waveform, which can be broken down into five separate frequency bands.

CHAPTER 2

LITERATURE SURVEY

2.1. OVERVIEW:

A summary of current studies in the area of motor imagery classification is given in this section. The focus is on feature extraction, channel selection, and classification—three crucial aspects that contribute significantly to the overall performance of MI systems. Identifying significant patterns in EEG data that correspond to various motor imagery tasks requires feature extraction. Channel selection methods are used to find the extremely pertinent EEG channels and eliminate noisy or redundant ones in order to improve efficiency and reduce computational complexity. Many studies have explored optimization-based approaches for this purpose. Based on the features that were retrieved, classification techniques are then applied to distinguish between different motor imagery states. A wide range of research efforts have been directed toward enhancing the accuracy, robustness, and real-time applicability of MI classification by integrating effective strategies across these stages.

In the area of BCI, MI has been extensively researched due to its potential in enabling communication and control without physical movement. Numerous studies [11] [12] [13] [14] [15] have been conducted to improve MI-based systems' performance. Early research concentrated on finding recurring patterns in EEG readings and comprehending how the brain reacts to imagined actions. In order to better capture these patterns, the emphasis gradually turned to better signal processing methods. To make it simpler to differentiate between several mental tasks, feature extraction techniques have been established to represent the EEG signal. In order to enhance classification accuracy and reduce system complexity, researchers have also realized how important it is to choose the most useful EEG channels. Various methods have also been explored for classifying MI tasks, aiming for higher precision and faster response times. Overall, the literature demonstrates a consistent attempt to improve the dependability, speed, and usability of MI-based structures, making them suitable for practical uses including neurofeedback, assistive technology, and rehabilitation.

2.2.   FEATURE EXTRACTION:

Numerous feature extraction approaches have been investigated in a number of research in the literature to improve motor imagery (MI) categorization efficiency. The aim of these approaches is to capture the most informative patterns from EEG signals, which are essential for precisely differentiating between various imagined actions. The following studies discuss various feature extraction techniques employed in motor imagery (MI) classification. Study [16] developed a new dual-stream convolutional neural network (DCNN) by classifying tasks using the time and frequency-domain elements of EEG signal. This network can take inputs as both time and frequency domain signals, and it combines the time and frequency domain extracted features by linear weighting for training. DCNN could also learn the weight on its own. The BCI II, III, and IV 2a dataset studies demonstrated that the model suggested in this works outperforms other standard methods. The mechanism that utilized time-frequency signals as inputs worked better than the model that solely utilized time or frequency domain signals. Compared to models that simply used one signal as input, the accuracy of classification was better for each individual.

To distinguish left- and right-hand imagery movement, they first extracted features characteristics from EEG motor actions utilizing Discrete Wavelet Transform (DWT), and then they categorized the EEG signal for application employing an Artificial Neural Network (ANN) [17]. A pair of feature vectors from beta rhythm are employed as input by the feed-forward neural network classifier. We noticed that using 3 input feature vectors—such as mean, peak power and standard deviation—improved our categorization performance by 80% as compared to 78% with 2 input feature vectors. Study [18] suggested that the optimum classification algorithm and feature extraction mechanism may be found by comparation of some most popular systems on a shared base dataset. DWT and cross correlation are two methods of feature extraction that researchers have also examined and compared. Following Five classifiers have been developed that are kernelled logistic regression (KLR), logistic regression (LR), probabilistic neural network (PNN), multilayer perceptron neural network (MLP), and least-square support vector machine (LSSVM). We used BCI IV-a dataset from the BCI-III as the fundamental database to test the classifiers. We used a 10-fold cross-validation (CV) technique to test the algorithms. From experimental findings, it has been observed that DWT with LSSVM classifier is superior to other procedures. Feature extraction mechanisms such as time domain, frequency domain, time-frequency-domain examination have been widely used in MI-related classification, each contributing to capturing specific signal characteristics. While these approaches have shown promising results, Common Spatial Pattern (CSP)

methods have often achieved superior performance. This is primarily because CSP effectively enhances the discriminative spatial features between different motor imagery classes by maximizing the variance difference across EEG channels. By portraying the original EEG signals into a new spatial domain, CSP highlights the most relevant patterns associated with imagined movements, leading to improved classification accuracy. Spatial features try to find features from certain electrode located on the scalp, like CSP. For MI-EEG data, the most prevalent ways to extract features are CSP and its derivatives. The EEG recognition method in [6] was built using DWT, CSP, and extreme learning machine (ELM). CSP and DWT are employed together to extract features, which fixes the issues that classical CSP was too sensitive to noise. Using ELM for categorization makes the BCI system work better in real time. Our results demonstrate that the algorithm works well because it has a categorization accuracy of 90% and a categorization time of 0.012 seconds for BCI III 2003 dataset.

A number of researchers have proposed enhancing and generalizing the CSP approach. The Study [7] study suggested a new sparse CSP approach for choosing channels in EEG signal. The suggested sparse common spatial pattern method was made to be an optimization method for picking the fewest channels although still keeping the accuracy of the classification high. So, the suggested method could be changed to get the finest possible categorization accuracy by getting rid of the noisy and extra channels, or to keep the fewest channels without lowering the categorization accuracy of employing all the channels. We tested the suggest sparse common spatial pattern method on two motor imaging databases, one with a small number of channels and one with a high number of channels. The suggested SCSP channel selection worked well in both databases to minimize the channels and was more accurate at classifying than other methods like mutual information, fisher criterion, SVM, CSP, and RSCP. The suggested SCSP algorithm significantly increased the accuracy of classification by a mean of 10% when compared to using three channels such as C3, C4, and C-z.

[8], introduced an approach named stationary Common Spatial Pattern (s-CSP), that normalizes the classical CSP algorithm in the direction of stationary subspaces. In their research, it was shown that their approach yields much better classification accuracy, especially for individuals who struggle to operate a BCI system. Based on this work, [9] reviewed the existing spatial filter calculation techniques and suggested a generic framework based on divergence maximization. In the framework, the standard CSP algorithm was re-described as a divergence maximization task, providing a principled means for the implementation of diverse invariances as well as for the incorporation of additional subjects' information. Not only does the framework encompass many CSP variations but also allow for new spatial filtering techniques through using diverse divergences and regularization settings. Building upon these

concepts, [10] furnished CSP with a probabilistic interpretation and introduced the P-CSP framework. P-CSP represents the EEG in the spatial-temporal domain and solves CSP's problem of over-fitting. Statistical inference techniques have been designed by the authors to overcome the problem of local optima. A computationally efficient algorithm for MAP estimation under isotropic noise was introduced through the use of Eigen decomposition. A variation inference solution was put forward for more complicated cases with the use of group-wise sparse Bayesian learning automatically adjusting the optimal model size.

Filter bank CSP (FBCSP) is an expanded variant of the traditional CSP method which makes use of both the frequency data in MI-EEG signals and the spatial information in EEG channels. FBCSP did the greatest job in classifying motor imagery out of all the other approaches which use manual feature extraction. Dataset 2a includes 4-class EEG data from 22 channels across 9 subjects, while Dataset 2b contains 2-class EEG data from 3 bipolar channels across 9 subjects. Multi-class extensions—Divide-and-Conquer (DC), Pair-Wise (PW), and One-Versus-Rest (OVR)—were used for Dataset 2a. Two feature selection methods, MIBIF and MIRSR, were applied to Dataset 2b. Utilizing 10×10-fold cv and session-to-session transfer, FBCSP achieved the highest performance, with mean kappa values of 0.569 (2a) and 0.600 (2b), outperforming other BCI Competition IV submissions.

2.3. OPTIMIZATION BASED CHANNEL SELECTION:

In order to enhance signal quality and lower computing complexity in BCI related systems, channel selection [19] [20] is an essential step. Through choosing the most relevant EEG channels, BCI systems can enhance classification accuracy and reduce noise. There are filter-based approaches [21] (like correlation or mutual information), wrapper-based methods [22] (like Recursive Feature Elimination), and embedding methods [23] (like Lasso regression) that can help you choose a channel. These algorithms find the channels that give the greatest information, which can be very important for things like classifying motor images. Choosing the right channels helps improve performance, making BCI systems more useful and efficient in the real world.

Optimization-based channel selection aims to identify the extremely beneficial EEG channels for motor imagery (MI) classification, improving both accuracy and computational efficiency. Metaheuristic algorithms, such as Genetic Algorithms (GA) [24], Particle Swarm Optimization (PSO) [25], and Grey Wolf Optimizer (GWO) [26], are commonly employed to choose the appropriate channels. These methods evaluate different channel subsets based on classification performance, enabling adaptive and subject-specific solutions. By refining the selection process, optimization techniques help to achieve higher accuracy while reducing computational complexity. Several studies have successfully applied

optimization-based channel selection to enhance MI classification, demonstrating its effectiveness in improving both performance and generalization across subjects. The following studies discuss various channel selection methods employed in MI classification.

Study [15] proposed the Logistic S shaped Binary Jaya Optimization Algorithm (LS-BJOA), that merges a Jaya optimization algorithm with logistic map initialization for reducing the computational complexity from multiple EEG channels. By introducing stochasticity, the logistic map improves predictive accuracy. The method initialized three electrodes as candidates and iteratively selects the most relevant channels using a bi-objective fitness function which maximizes accuracy and minimizes selected channels. The signals were preprocessed with a fifth-order bandpass filter and ICA for artifact reduction, Regularized Common Spatial Pattern (RCSP) used for feature extraction. Classification was performed using SVM, Linear Discriminant Analysis (LDA) and Naïve Bayes (NB). Tested on three public EEG datasets, their technique accomplished categorization accuracies of 83.59%, 89.02% and 82.09% with fewer channels, while reducing computational time. A MX-BBOA Algorithm named as Multi objective X shaped Binary Butterfly Optimization, was proposed in a study [27] for effective channel selection in motor imagery (MI) categorization. The technique mimics how butterflies forage for food in order to determine which channels are the most instructive. First, pertinent frequency responses are captured using a 5th-order Butterworth bandpass filter. To remove artifacts the Independent Component Analysis (ICA) was used. Temporal-Spatial features are subsequently collected from the improved data signals employing the Multivariate Empirical Mode Decomposition (MEMD) method. Two sigmoid transfer functions are used to minimize the search space of continuous channel to a search space of binary in order to increase the diversity of the solutions. SVM was employed to classify the collected features in order to differentiate MI task pairs, including right hand, feet, left hand, and tongue. Validation on 3 available public EEG datasets (BCI IV-Dataset 1, BCI IV-IIA, and BCI III-Dataset IV-a) demonstrates that their technique accomplishes superior categorization accuracies of 79.74%, 84.49%, and 84.55%, respectively, with fewer channels compared to state-of-the-art methods.

In [17], introduced an optimization algorithm using the Fisher score for channel selection in EEG. Firstly, they derive the CSP features of the EEG data in various bands, compute each channel's fisher score using these features and rank them. Then, they utilized an optimization algorithm for the final selection of the channels. On the BCI IV 2a dataset, their algorithm chooses on average 11 channels out of 4 bands with a mean accuracy of 79.37%. It is a development of 6.52% over the use of the entire 22 channels. Their approach achieved comparable outcomes on their own dataset, which they recorded, with a mean accuracy of 76.95% and an increase of 24.20% using fewer than half of the channels. Nonetheless, the

optimization approach of channel selection can have limitations, for instance, in excluding the most crucial channel because of initialization using the wrong channels and not providing ideal performance because to the absence of an effective multi objective fitness function. To overcome these drawbacks, [18] introduced the MPJS (multi-objective prioritized jellyfish search algorithm) that has two major improvements. Firstly, domain specific initialization was utilized to choose the extremely crucial channels at the process of initialization in order not to exclude some important channels. Second, choosing the most helpful corresponding channels using a multi-objective fitness function rather than a single-objective one makes sure that the chosen channels satisfy the number requirements while including candidate channels. They developed an enhanced double-branch EEGNET (DB-EEGNET) to attain the best results in 4-class Motor imagery detection. Their proposed MJPS channel selection and classification model DB-EEGNET algorithm outperformed the baseline approach on the BCI IV-IIA dataset, BCI IIIA dataset, and high gamma datasets, achieving accuracies of 83.9%, 84.46%, and 94.78%, respectively.

Feature extraction [28] is a crucial stage in the classification of motor imagery (MI) tasks because it transforms unprocessed EEG signals into meaningful representations. The first step in the procedure is pre-processing the raw EEG data, which frequently include bandpass filtering to separate frequency bands [29] like Alpha (8–12 Hz) and Beta (12–30 Hz), which are frequently linked to motor imagery. To capture the underlying brain activity, different features are retrieved from the pre-processed data. Overall, these attributes can be splitted into 3 categories: time-domain [30] (like mean & standard deviation), frequency-domain (like band power and power spectral density [31] (PSD)), and time-frequency (using wavelet transform or short-time Fourier transform [32] (STFT)). In this work we used FBCSP (Filter bank Common Spatial Pattern) [33] and AGSP (Advanced Graph Signal Processing) for feature extraction [34].

2.3.1. Filter Bank Common Spatial Pattern:
An enhancement of the conventional Common Spatial Pattern (CSP) technique, Filter Bank Common Spatial Pattern (FBCSP) [33] [35] incorporates frequency-specific information to enhance feature extraction from EEG signals. By learning spatial filters that optimize the variance differences between 2 classes, CSP effectively improves the discriminability of EEG features. However, because it only uses one frequency band, it might not be able to capture the complex frequency-dependent dynamics of brain activity. In order to overcome this restriction, FBCSP breaks down EEG signals into a number of different frequency bands utilizing a filter bank made up of numerous bandpass filters. In BCI applications, FBCSP has demonstrated highly beneficial, particularly for motor imagery classification, where subject-specific discriminative patterns may exist in many frequency bands depending on the subject and

the task. By leveraging multi-band decomposition, FBCSP improves robustness and classification accuracy, making it a standard approach in state-of-the-art EEG-related systems.

2.3.2. Advanced Graph Signal Processing:

An expansion of conventional Graph Signal Processing (GSP), Advanced Graph Signal Processing (AGSP) uses the graph representation of signals to do more complex and structured analyses. AGSP involves the application of graph-based techniques to model, process, and analyze signals that are inherently structured or relational in nature, such as those coming from brain networks in BCI systems. Key concepts of AGSP given below:

- **Graph Representation:** AGSP uses a graph to describe data, with nodes standing for **signals** or data points and edges for dependencies or relationships between nodes. Each electrode (or set of electrodes) in an EEG can be represented as a node in the graph, and the connections between channels—such as those based on functional connectivity or spatial proximity—are modelled as edges.

- **Signal Processing on Graphs:** AGSP aims to process these graph-based signals by using techniques like graph convolution, graph Fourier transform (GFT), and **spectral** graph theory. This allows the incorporation of structural information, such as brain connectivity, when processing the signals.

- **Graph-based Features:** AGSP techniques can record functional connectivity across brain regions in the context of EEG and motor imagery, illustrating the relationship between activity in one area of the brain and **activity** in another. Features derived from these graph-based representations can include measures of centrality, clustering, and community structure, that can then be used to classify different mental states.

- **Advantages:** AGSP provides a way to process and analyse signals that have an inherent network or graph structure. This is particularly useful in neuroscience and BCI where brain activity is often interconnected, and modelling **the** brain as a network can provide richer insights than traditional signal processing methods.

2.4. CLASSIFICATION:

In systems that use motor imagery as a MI-BCI, classification was essential. Following the extraction of significant features from raw EEG signal, the categorization stage entails differentiating between various motor tasks, such as visualizing the movements of the tongue, left hand, right hand, or feet. The goal is to accurately map the extracted neural patterns to the corresponding motor imagery classes. The BCI system's performance heavily relies on the effectiveness of the classifier used, as it

directly impacts the system's ability to interpret user intentions in real-time. Various classifiers, ranging from traditional machine learning models to advanced ensemble and deep learning techniques, have been explored to enhance classification accuracy and generalization across subjects and sessions. The following studies discuss various classification methods employed in motor imagery (MI) classification.

Study [19], employed ensemble classifier of optimized ELM. PSO was used for optimizing the ELM's hidden biases and input weights in order to overcome the Classification instabilities performance and randomness in case of using randomly generated parameters of the ELM and the categorization performances of the ensemble of base classifiers are combined using the majority voting technique in order to overcome the detrimental effect of the classification performance due to the use of locally optimal parameters of the ELM. Utilized two publicly available EEG datasets along with some previously published methods from the literature, the suggested approach was tested against four competing approaches. The outcomes suggested that the presented approach performed better than competitive approaches with notably higher categorization accuracies on both four-class and two-class MI data. Furthermore, in contrast to the current techniques, it also maintained higher mean accuracy performance of two-class categorization and achieved higher for the subjects with the lower accuracy performance on both four-class and two-class categorizations. Several end-to-end deep learning (DL) architectures have replaced handcrafted features in Motor Imagery EEG decoding in recent years. In addition, most of existing DL architectures are designed for the compensation of the limitation of traditional BCI mechanism and the promotion of Motor Imagery EEG decoding performance. CNN was commonly used in Motor imagery EEG classification due to its capable for learning informative features from the dataset. Despite the fact that these DL based methods performed better than traditional ML approaches, their advancements are limited and spatial or temporal variables are not completely utilized. Recently, various well-thought-out DL architectures was created to fully extract Motor imagery EEG signals' multi-domain information and improve decoding performance. [20] created a multi-branch three-dimensional CNN for the new representation and developed a three-dimensional architecture of the motor imagery EEG data to preserve the spatial detail of the sampling electrodes. To integrate the spectral -temporal features and boost the accuracy of Motor imagery EEG classification, a temporal-spectral-based squeeze-and-excitation feature fusion network (TS-SEFFNet) had been suggested in 2021 [36] . Nevertheless, such networks only extract features from a specific time window of motor imagery EEG signals, disregarding the significance of motor imagery-based patterns throughout time, which contributes to poor classification performance. A variance layer was recently suggested by FBC-Net [37] for effectively obtaining features from multiple windows of the time series, and it accomplished extremely well on publicly available datasets. However, FBC-Net does not explore

temporal relationships among features; instead, it explores features in multiple time windows independently, which makes it insufficient for discriminative extraction of features.

Study [23] suggested an ideal channel selection method for multiclass Motor imagery categorization via a fusion convolutional neural network with attention blocks (FCNNA). The authors established a CNN based model by combining layers of a convolutional blocks with various both temporal and spatial filters. Both spatial and temporal filters are especially developed to identify the signal distribution and correlations of signal characteristics amongst electrode locations and toward analyse how these characteristics change over time. To increase the feature extraction capability of the EEG signals, the Convolutional Block Attention Module (CBAM) was applied after these layers. While channel selection, the best combination of channels has been selected using a new approach with the help of the genetic algorithm to provide variable channels as well as fixed for all the subjects. The suggested methodology was verified with 6% increase in multiclass classification accuracy against majority of baseline models. Remarkably, we attained highest accuracy of 93.09% for the left-hand and right-hand movement binary classes. Additionally, the cross-subject approach for multiclass categorization achieved 68.87% accuracy. After channel selection, accuracy of multiclass classification was improved and reached 84%. Recent MI-BMI-net [24] proposed an automatic channel selection approach of EEG channels in terms of spatial filters for choosing most informative EEG channels towards Motor Imagery EEG decoding. In contrast, we focus on emphasizing the significance of temporal relationships between features in separate time intervals and attain additional accuracy of 2.3% compared to MI-BMI-net. These evidences suggest learning the temporal relationships between features of separate time intervals can enable capture of discriminative features as well as enhance performance of MI-EEG decoding.

Traditional classifiers like K-Nearest Neighbors (KNN) [38] and Support Vector Machine (SVM) [39] are frequently used in EEG-related motor imagery classification because of their effectiveness and simplicity. To categorize patterns from EEG signals, these algorithms either use distance measures or learn decision boundaries. They may, however, find it difficult to handle the high dimensionality, noise, and fluctuation that are frequently found in EEG data. On other hand, ensemble classifiers including as Random Forest [40] (RF), XG-Boost [41], and Ada-Boost [42] combine the predictive capabilities of several base models to improve overall performance.

CHAPTER 3

PROBLEM STATEMENT & DEVELOPMENT PROCESS

3.1. PROBLEM STATEMENT

Accurate classification in MI-related BCI systems remain a challenging task because of the complexity, non-stationary behavior of EEG data and the presence of redundant or noisy channels. Traditional feature extraction and channel selection approaches frequently not be able to gather the underlying spatial and functional connectivity patterns crucial for decoding MI tasks. To address this, advanced signal processing and optimization strategies that can improve feature quality and channel matching are needed. In this work, we aim to extract the most discriminative and physiologically meaningful features using advanced Graph Signal Processing (AGSP), which effectively models the spatial relationships among EEG electrodes.

Furthermore, to enhance the robustness and efficiency of the channel selection process, we incorporate Set-based Integer-coded Fuzzy Granular Evolutionary algorithm. SIFE mechanism for intelligent channel selection. SIFE utilizes swarm intelligence to discover the search space and select the most informative subset of EEG channels, effectively reducing dimensionality while preserving key discriminative features. This not only boosts categorization accuracy but also minimize computational cost and enhances the system's real-time capabilities. This approach minimizes the dimensionality of the data and enhances classification accuracy by focusing on the most informative EEG channels.

3.2. DEVELOPMENT PROCESS:

3.2.1. REQUIREMENT ANALYSIS:
Requirements are a system characteristic or definition of some activity that the system is able to do to satisfy the purpose of the system. It gives a mechanism for understanding correctly what is required by and wanted by the customer, for critiquing against need, determining feasibility, agreeing upon a feasible solution, describing the solution clearly, and checking and maintaining requirements as they get converted into an operating system.

3.2.2. PYTHON:

Python is known for its extensive features that make it a versatile and powerful programming language. One of its standout features is the simple and readable syntax, which emphasizes clarity, making it easy to learn and use even for beginners. Python is an interpreted programming language that runs code line by line, permitting rapid debugging and testing without requiring for compilation. It is also dynamically typed, meaning you do not need to declare variable types explicitly; they are assigned at runtime, enhancing flexibility. Python's compatibility with multiple platforms enables it to execute effortlessly on different platforms such as Windows, macOS, and Linux. Additionally, it boasts a vast ecosystem of public library and frameworks (including Pandas for data analysis, NumPy for numerical computations, and Tensor Flow for machine learning), making it appropriate for a widespread range of applications, from data science to web development automation. These features have contributed to Python's popularity in both academia and industry.

3.2.3. GOOGLE COLAB:

Google Collab is a cloud-based Jupyter Notebook service offered by Google for free, intended for data science, machine learning, and deep learning applications. Google Collab makes powerful computing assets accessible to its users easily with GPUs and TPUs, accelerating computation significantly without incurring costly hardware costs. Collab is hosted completely in the cloud and does not require set-up; therefore, its users can start coding straight out of their web browsers. Collab is integrated with Google Drive so easily that file storage, access, and sharing become uncomplicated. Collab is pre-loaded with common Python packages such as NumPy, Pandas, Tensor Flow, and PyTorch so that it is all set for different data science workflows. Collab is also capable of real-time collaboration so that multiple people can co-work on a single notebook simultaneously as is the case with Google Docs. With its convenience, affordability, and flexibility, Google Collab is a valuable utility for data science and machine learning for both amateurs and experts alike.

3.2.4. RESOURCE REQUIREMENTS FOR GOOGLE COLAB:
3.2.4.1. Internet Connection:

Since Google Collab is a cloud-related service, accessing and using notebooks requires a consistent internet connection.

3.2.4.2. Google Account:

To use Collab, a Google account is required. This allows access to Google Drive for storing notebooks and data files.

3.2.4.3. Browser:

Collab works on any current web browser, such as Mozilla Firefox, Google Chrome, or Microsoft Edge. It is optimized for these browsers and provides a smooth user experience.

3.2.4.4.   **Memory (RAM):**

Collab provides 12 GB of RAM for free-tier users, with the option to upgrade to 25 GB of RAM in some cases. However, this is dependent on the availability of resources, and heavy computations may require request more memory.

3.2.4.5.   **CPU/GPU/TPU:**

For standard tasks, Collab uses CPU resources. For more intensive computations, you can access NVIDIA GPUs and Google TPUs (limited usage per day for free users). We can switch between CPU, GPU, and TPU through the runtime settings.

3.2.4.6.   **Storage:**

Collab offers storage via Google Drive for your files, datasets, and notebooks. Free Google Drive accounts provide 15 GB of storage, with additional storage available through paid plans.

3.2.4.7.   **Software Libraries:**

Collab supports various Python libraries by default, including NumPy, Pandas, Matplotlib, and PyTorch. We can also install additional libraries using pip or conda.

CHAPTER 4

DATASET DETAILS

4.1.   DETAILED DESCRIPTION OF BCI IV 2A DATASET:

There are 9 participants in this data set [25] who had EEG recordings. Four distinct motor imaging assignments were included in the cue-based BCI approach: class 1 involved visualizing moving the left hand, class 2 involved the right hand, class 3 involved both feet, and class 4 involved the tongue. Two sessions were recorded by each individual on two separate days. There were short breaks between each of the six runs. There are 48 trials in each run, twelve for each of the four different classes. This means that there are 288 trials in each session. To out how much the electrooculogram (EOG) affected the session, a recording of approximately 5 minutes was produced at the start of each. There are three sections to the recording: 1. Observing a fixation cross on the screen for 2 minutes with the eyes open, 2. 1 minute with eyes closed, & 3. 1 minute with eye activities. Figure 4.1 shows the time plan for one session. The participants were seated in an armchair facing a screen. At the starting point of the trial, the fixation cross displayed on the dark screen (t = 0 second). There is also a short sound warning tone that plays.

.
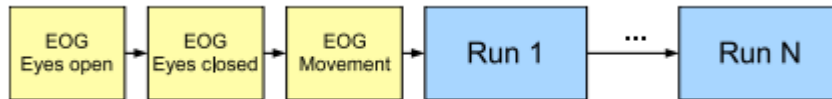
Figure 4.1. Timing plan of one session

After t = 2 s, an arrow pointing up, down, left, or right direction displayed on the computer screen for 1.25 seconds and told the subjects to do the target motor imagery action. One of the four motor imagery classes was displayed by the arrow: 1. Left hand, 2. right hand, 3. foot, or 4. tongue. There was no feedback provided. The individuals were told to do the MI action till t = 6 seconds, if the fixation cross on the computer screen

disappeared. After that, there was a short gap during which the screen went black again. Figure 3.2 shows the model.
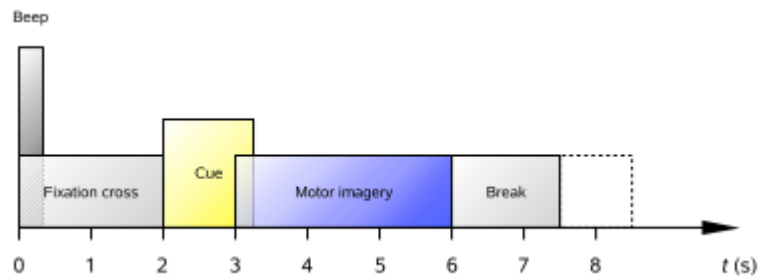
Figure 4.2. The paradigm's timing scheme

4.2. DATA RECORDING:

The EEG signals was captured with 22 Ag/Ag-Cl electrodes that were 3.5 cm apart. Figure 3.3 on the bottom shows how they were set up.
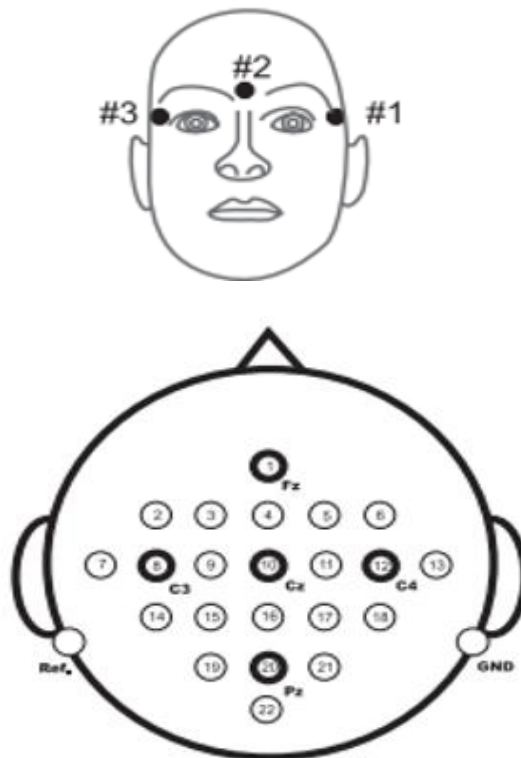
Figure 4.3: Top: Montage of the three monopolar EOG channels' electrodes

Bottom: Montage of electrodes in accordance with the international 10-20 system

The bottom mastoid was employed as a reference point whereas the top mastoid was employed as ground for the monopolar recording of the EEG signals. These signals were captured at 250 Hz and filtered to only let through frequencies range between 0.5 Hz and 100 Hz. The amplifier's sensitivity has been configured to 100 μV. There was also a notch filter with 50 Hz on to get rid of line noise. Apart from the 22 EEG channels, there were also 3 monopolar channels for vertical EOG that were gathered and it was sampled at 250 Hz (see the right side of Figure 4.3). The 3 EOG channels are not to be employed for categorization; rather, they are included for the later application of artefact correction techniques.

Table 4.1: List of event types (the 1st column encompasses hexadecimal values and the 2nd column encompasses decimal values).

| Description	Event
Eye open	276
Eyes closed	277
Start of a new trial	768
Class 1 (Cue onset left hand)	769
Class 2 (Cue onset right hand)	770
Class 3 (Cue onset foot)	771
Class 4 (Cue onset tongue)	772
Unknown Cue	783
Rejected trial	1023
Eye movements	1072
Start of a new run	32766

The workspace will hold two key components: a signal array s and a header structure h. Signal data includes 25 channels—22 for EEG records and the remaining 3 for EOG records. The header structure provides event-related data that outlines how the information evolves throughout time. Three main fields in the header are essential for evaluating the dataset:

1. The type of evert – h.EVENT.TYP

2. The sample position of the event – h.EVENT.POS

3. The during of event– h.EVENT.DUR

The event position is recorded in h.EVENT.POS, while h.EVENT.TYP indicates the event type, and h.EVENT.DUR shows how long the event duration lasts. Event types used in this dataset are given in both hexadecimal and decimal formats. It's important to note that the class labels (1 to 4 for event types 769 to 772) are available only in the training data. Trials identified by experts as containing artifacts are labelled with event type 1023. Additionally, h.Artifact Selection having lists of each trial as either clean (0) or containing an artifact (1).

CHAPTER 5

PROPOSED ALGORITHM FOR MOTOR IMAGERY CLASSIFICATION

5.1.   OBJECTIVE:

- To use Advanced Graph Signal Processing (AGSP), which is specifically designed for motor imagery classification, to extract discriminative and informative features from EEG signals

- To use the Set-based Integer-coded Fuzzy Granular Evolutionary (SIFE) channel selection algorithm to pick the best EEG channels, improving model accuracy and efficiency by minimizing redundancy

- To ensure reliable and broadly applicable MI classification findings by enhancing classification performance with ensemble classifiers like XG-Boost, Ada-Boost, and Random Forest.

5.2.   WORKFLOW OF PROPOSED METHODOLOGY USING FBCSP-PSO:

In this section, we discuss the classification strategy employed for BCI MI tasks. This approach integrates FBCSP for effective feature extraction and PSO for optimal channel selection. The combination of FBCSP and PSO improves the discriminative ability of the extracted features while reducing redundancy by selecting the most informative EEG channels. This optimized feature set is then used to train various classifiers, enabling accurate identification of motor imagery classes and ensuring improved performance in BCI applications.

Figure 5.1 shows general block diagram of FBCSP-PSO methods. Initially, the EEG dataset underwent preprocessing to ensure data quality and suitability for feature extraction. Since FBCSP was utilized for feature extraction, the data was decomposed into nine distinct frequency bands: 4–8 Hz, 8–12 Hz, 12–16 Hz, 16–20 Hz, 20–24 Hz, 24–28 Hz, 28–32 Hz, and 32–40 Hz. Then, CSP algorithm [43] was employed to each band to extract discriminative spatial features corresponding to motor imagery tasks. Following feature extraction, channel selection was performed using

the PSO algorithm to reduce dimensionality and retain the most appropriate EEG channels for categorization. The optimized feature set was then classified using machine learning approaches such as Support Vector Machine (SVM), Random Forest (RF) and K-Nearest Neighbors (KNN), to evaluate and compare their efficiency in identifying motor imagery intentions.
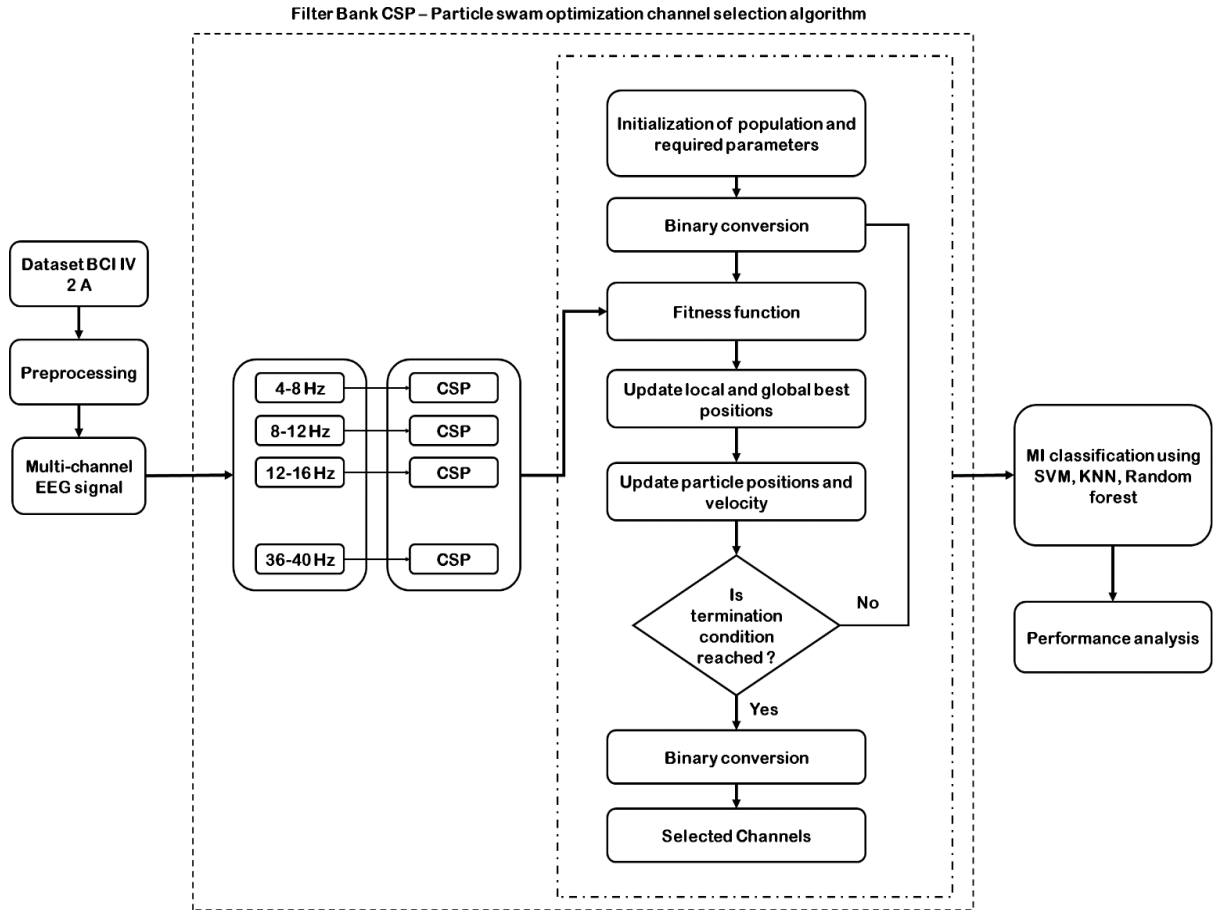
Figure.5.1. General Diagram of the FBCSP-PSO method

5.2.1. FILTER BANK COMMON SPATIAL PATTERN (FBCSP):

FBCSP [44] is an advanced feature extraction technique widely used in BCI related applications, especially for motor imagery classification. It extends the CSP technique by applying it across numerous frequency sub-bands using a filter bank, thus capturing richer discriminative information from different brain rhythms. Working procedure was given below:

5.2.1.1. Band-pass Filtering (Filter Bank Stage):

EEG signals contain multiple oscillatory components corresponding to distinct brain activity tasks. In this stage, the raw EEG data is passed via a filter bank consisting of multiple band-pass filters to separate it into distinct frequency bands. For example, the signal is split into 9 overlapping or non-overlapping bands: [4 − 8] , [8 − 12] , [12 − 16] , [16 − 20] , [20 − 24] , [24 − 28] , [28 − 32] , [32 − 36] , [36 − 40] Hz [4–8],[8–

12],[12–16],[16–20],[20–24],[24–28],[28–32],[32–36],[36–40] Hz   Each band focuses on different brain rhythms (like theta, alpha, beta), enhancing the extraction of frequency-specific information.

5.2.1.2. Common Spatial Pattern (CSP) for Each Band:

For each filtered sub-band, the Common spatial pattern method is used to construct spatial filters which maximize variation for one class and minimize for the other. Let $X_b \in \mathbb{R}^{C \times T}$ be the EEG signal in the bth band (C = channels, T = time points). The spatial filters W_b are tackling a general eigenvalue challenge:

$$R_1 W = \lambda R_2 W \tag{5.1}$$

Where R_1 and R_2 are covariance matrices of the two different classes. The resulting matrix W contains spatial filters that project the EEG data into a new space with maximized class separability.

5.2.1.3. Feature Computation:

After projection, the EEG signal is transformed as:

$$Z_b = W_b^T X_b \tag{5.2}$$

The feature for each component is computed using log-variance:

$$f_i = \log\left(\frac{var(Z_{b,i})}{\sum_{j=1}^m var(Z_{b,i})}\right) \tag{5.3}$$

Where $Z_{b,i}$ is the ith component of the projected signal in band b, and m is the number of spatial filters.

5.2.1.4. Feature Concatenation Across Bands:

Once features are extracted from all possible bands, they are concatenated to create the final feature vector:

$$F = [f^1, f^2, f^3, \ldots f^B] \tag{5.4}$$

Here, f^b is the feature vector from the bth frequency band, and B is the total number of bands.

5.2.2. PARTICLE SWARM OPTIMIZATION CHANNEL SELECTION ALGORITHM:

In BCI related applications, not all EEG channels contribute equally to classification. Some may contain noisy or redundant information. Hence, channel selection is crucial to minimize dimensionality, enhance categorization accuracy, and reduce computational cost. PSO is the one effective method to address channel selection problem. Algorithm 4.1 represent the pseudocode of Particle Swarm Optimization (PSO) channel selection algorithm.

Algorithm 5.1. PSO channel selection algorithm

Input: Total number of channels: N, Population size: P, Inertia weight: w, Maximum iterations: max_iter, Social and cognitive constants: c1, c2, FBCSP features extracted data

Output Optimal channel subset

Begin
  Initialize swarm with P particles
  For each particle:
    Randomly initialize binary position vector of length N
    Initialize velocity vector
    Evaluate fitness using classifier and store as personal best (pBest)
  Identify the global best particle (gBest) with highest fitness
  For iter = 1 to max_iter do:
    For each particle i in swarm:
    For each dimension j in particle:
      Update velocity:

$$[i][j] = w * v[i][j] + c1 * rand1 * (pBest[i][j] - position[i][j]) + c2 * rand2 * (gBest[j] - position[i][j])$$

      Apply sigmoid to get probability:
      $prob = 1 / (1 + exp(-v[i][j]))$
      Update position (binary):
        If rand () < prob:
          $position[i][j] = 1$
        Else:
          $position[i][j] = 0$
    Evaluate new fitness:
     If current fitness > pBest fitness:
       Update pBest[i] = current position
     If current fitness > gBest fitness:
       Update gBest = current position
  Return gBest (optimal selected channels)
End

In this study, PSO [45] is utilized as a channel/feature selection method to identify the extremely beneficial EEG channels that contribute to effective motor imagery classification. Motivated by social habit patterns of fish schooling and bird flocking, PSO works by initializing a population of particles where each search agent signifies a possible solution—in this case, a binary vector indicating the selection status of EEG channels. A value of '1' signifies that the channel is selected, whereas '0' indicates it is not. Each particle explores the search space and is evaluated using a fitness function, which is primarily based on the categorization accuracy achieved

utilizing only the selected EEG channels. To enhance efficiency, the fitness function can also incorporate a penalty term to minimize the selected number of channels, thus balancing performance and computational cost.

During each iteration, each particles update their positions and velocities using current best-known position (p-Best or personal best) and the global best position (g-Best) discovered by any particle in the swarm. The velocity updated by three components: inertia, cognitive influence, and social influence. The new positions are determined using a sigmoid function that transforms continuous velocity values into binary decisions, enabling the particle to select or deselect individual channels. This process continues iteratively, with the swarm converging towards an optimal subset of channels that yield the highest classification performance. By selecting only, the most relevant channels, PSO not only reduces dimensionality but also accelerates the classification process and improves model generalization. The main advantages of PSO include its global search capability, compatibility with any classifier (such as SVM, KNN, or RF), and its ability to achieve high accuracy with fewer features. We used SVM classifier. However, its performance is sensitive to parameter tuning and the computational cost may be significant due to repeated classifier evaluations for each particle across generations.

5.3. WORKFLOW OF PROPOSED METHODOLOGY USING AGSP-SIFE ALGORITHM:

In this section, we discuss the classification strategy employed for BCI MI tasks. The approach integrates AGSP (Advanced Graph Signal Processing) for effective feature extraction and SIFE (Set-based Integer-coded Fuzzy granular Evolutionary) [46]) for optimal channel selection. The combination of AGSP and SIFE improves the discriminative ability of the extracted features while reducing redundancy by selecting the most informative EEG channels. This optimized feature set is then used to train various classifiers, enabling accurate identification of motor imagery classes and ensuring improved performance in BCI applications.

Initially, the dataset was pre-processed to extract EEG data components in the frequency range between 4 to 40 Hz, which are known to capture key neural oscillations relevant to motor imagery tasks. Following this, an advanced signal graph processing technique was employed to convert the filtered EEG signals into graph representations. These graph signals were then processed to extract a rich set of statistical features that encapsulate both spatial and topological characteristics of the EEG signal. After feature extraction, channel selection was performed utilizing the SIFE method to find the extremely appropriate channels contributes to accurate categorization. Finally, the selected features were used for classification using an ensemble ML classifier, such as Extreme Gradient Boosting (XG-Boost), Ada-Boost and Random Forest (RF). These ensemble methods

enhance the reliability and accuracy of the ML model by combining multiple learners to make final decisions. The General diagram of the AGSP-SIFE channel selection method given in Figure 4.2.
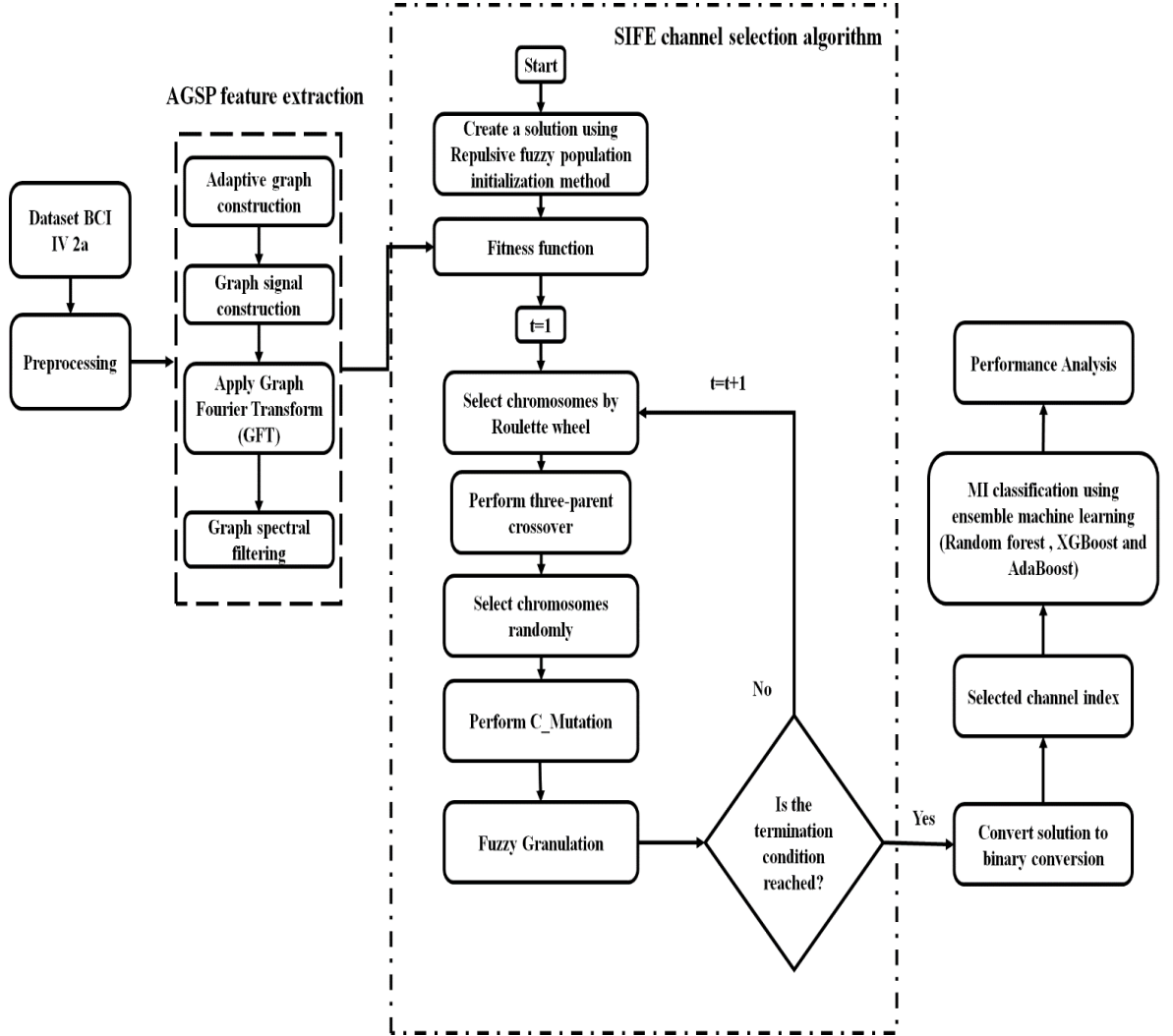
Figure.5.2. General Diagram of the AGSP-SIFE channel selection method

Benefits of Extracting EEG Signals in the 4–40 Hz Range: The frequency range between 4–40 Hz encompasses several important EEG bands that are highly appropriate for MI tasks. These include the theta (4–8 Hz), alpha (8–13 Hz), beta (13–30 Hz), and low gamma (30–40 Hz) bands. Each of these bands contributes uniquely to brain activity while motor imagination and planning:

Table 5.1. Bands contributes uniquely to brain activity during motor planning and imagination:

Theta θ (4–8 Hz)	Linked with cognitive processing and sensorimotor integration, which can play a role in MI tasks involving attention and mental effort.
Alpha α (8–13 Hz)	Typically observed while relaxed wakefulness, alpha rhythms, especially in the sensorimotor cortex (mu rhythm), are known to desynchronize during MI, indicating brain activation.
Beta β (13–30 Hz)	Linked with motor control and active concentration. Beta rhythms show characteristic event-related synchronization (ERS) and event related desynchronization (ERS) while motor imagery and execution.
Low Gamma γ (30–40 Hz)	Reflects higher-order cognitive processes and is often involved in sensorimotor coordination, making it valuable for enhancing MI-related feature representation.

5.3.1. ADVANCED GRAPH SIGNAL PROCESSING (AGSP):

Graph Signal Processing (GSP) [47] is an emerging framework that extends traditional signal processing to data defined over irregular domains such as graphs. In the context of EEG data examination, GSP is particularly powerful due to the spatially distributed nature of brain data captured from multiple electrodes positioned on the scalp. Each electrode may be considered a node in a graph, and edges signify spatial, functional, or statistical relationships between them. By employing GSP, signals may be examined not just in time & frequency domains, but also in the graph spectral domain, revealing insights about spatial dynamics and inter-electrode connectivity patterns.

5.3.1.1. Adaptive Graph Construction:

In Graph Signal Processing (GSP), adaptive graph construction refers to dynamically generating the graph topology based on data-driven relationships between signal nodes, instead of depending on exclusively on fixed anatomical or geometric layouts. For EEG signals, each channel or electrode is considered a node in the graph, and edges show connectivity or correlation. Adaptive graphs improve accuracy by capturing context-specific interactions between electrodes. Steps in Adaptive Graph Construction:

- **Step 1: Define Nodes**: Each and every EEG channel is treated as a node v_i in the graph

$$G = (V, E) \tag{5.5}$$

- **Step 2: Compute Edge Weights**: Edges $e_{i,j}$ are formed based on pairwise similarity or functional relationships:

$$w_{i,j} = \exp\left(-\frac{(x_i - x_j)^2}{\sigma^2}\right) \qquad (5.6)$$

- **Step 3: Form the Adjacency Matrix A**: A symmetric matrix representing pairwise edge weights.

- **Step 4: Compute Graph Laplacian L**: Either combinatorial or normalized:

$$L = D - A \qquad (5.7)$$

Where D is the degree matrix.

5.3.1.2. Graph Signal Construction:

Once the graph is constructed, EEG data is represented as a graph signal:
- Let $x \epsilon \mathbb{R}^N$ be a signal defined on the graph, where x_i is the value at node i (e.g., EEG amplitude at a channel).

- Each time segment or epoch from EEG is treated as a separate signal on this graph.

This process converts spatial EEG data into a form suitable for graph-based spectral analysis.

5.3.1.3. Applying Graph Fourier Transform (GFT):

The Graph Fourier Transform [48] enables transformation of graph signals into the spectral domain based on graph Eigen structure. Steps were given below:

1. **Eigen Decomposition of Laplacian:**

$$L = U \Lambda U^T \qquad (5.8)$$

Where:

- U is the eigenvectors matrix (graph Fourier basis)
- Λ is the eigenvalues diagonal matrix (graph frequencies)

2. **Graph Fourier Transform**:

$$\hat{x} = U^T x \qquad (5.9)$$

Transforms the spatial signal x into spectral coefficients $\hat{x}$, indicating how the signal varies over graph frequencies.

5.3.1.4. Graph Spectral Filtering:

In this study, graph spectral filtering is employed to enhance feature extraction from EEG signals by leveraging the heat kernel filter [49], a powerful smooth spectral filter inspired by the diffusion of heat over a graph. This approach enables the suppression of noisy or non-informative high-frequency components while preserving meaningful structural and

spatial characteristics of the EEG data. The heat kernel filter is a low-pass filter that models the diffusion of heat across a graph. It emphasizes smooth variations in graph signals, making it particularly effective for EEG, where neighboring electrodes often exhibit correlated activity. The heat kernel is stated in the spectral domain as:

$$g(\lambda) = exp(-\tau\lambda) \qquad (5.10)$$

Where: λ is the graph Laplacian eigenvalue (graph frequency), τ is the diffusion time parameter (controls the degree of smoothing), $g(\lambda)$ is the spectral response of the filter.

5.3.1.5. Statistical feature extraction:

After filter signal extraction we calculated the mean, standard deviation (SD), variance (Var), root mean square (RMS), skewness, and kurtosis [50]. These statistical features provide understandings into the distribution, central tendency, variability, and shape of the EEG signals, helping to describe the underlying temporal patterns effectively.

- **Mean**: The signal's average [51] value, or mean, gives an indication of its central tendency. The presence of specific cognitive states or variations in brain activity can be indicated by changes in mean values in EEG analysis,

$$Mean = \frac{1}{N}\sum_{n=1}^{N} S_n \qquad (5.11)$$

Where the total number of data points is N, and each data point is represented by S_n

- **Variance (Var):** Variance [51] quantifies how much the data points deviate from the mean, revealing the variability of the signal. A signal with low variation is steadier and more reliable, whereas one with high variance may imply erratic signal fluctuations or higher brain activity.

$$Variance = \frac{1}{N}\sum_{n=1}^{N}(S_n - Mean)^2 \qquad (5.12)$$

Where S_n is each data point, and $Mean$ is the mean value of the signal.

- **Standard deviation (SD) :** Standard deviation [50] measures how much the signal varies or disperses. The variance's square root is what it is.

$$std = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(S_n - Mean)^2} \qquad (5.13)$$

It measures the degree of deviation of the signal values from the mean. In EEG, higher standard deviation values can indicate more dynamic brain activity, such as during periods of heightened cognitive or emotional engagement.

- **Root mean square (RMS):**It gives a measurement [52] of the signal's strength, indicating its total magnitude. It is particularly useful in quantifying the strength of the EEG signal.

$$rms = \sqrt{\frac{1}{N}\sum_{n=1}^{N} S_n{}^2} \qquad (5.14)$$

Higher RMS values typically indicate stronger brain activity, and it can be used to differentiate between active and resting states.

- **Skewness**: Skewness [53] quantifies the distributional asymmetry of the signal. It helps in understanding whether the data points are more concentrated on one side of the mean.

$$Skewness = \frac{\frac{1}{N}\sum_{n=1}^{N}(S_n - Mean)^4}{\left(\sqrt{\frac{1}{N}\sum_{n=1}^{N}(S_n - Mean)^2}\right)^2} \qquad (5.15)$$

When skewness is negative, the left tail is longer, and when it is positive, the right tail is longer. This feature is useful for detecting irregularities or shifts in brain action patterns, that can be related to specific mental states or cognitive processes.

- **Kurtosis**: Kurtosis [54] measures the "tailed-ness" or the peaked-ness of the signal's distribution. It helps in identifying the presence of outliers or extreme values in the data.

$$Kurtosis = \frac{\frac{1}{N}\sum_{n=1}^{N}(S_n - Mean)^3}{\left(\sqrt{\frac{1}{N}\sum_{n=1}^{N}(S_n - Mean)^2}\right)^3} \qquad (5.16)$$

High kurtosis suggests the presence of sharp peaks and potentially abnormal brain activity, while low kurtosis indicates a flatter, more evenly distributed signal. It is useful in detecting epileptic spikes or other abnormalities in the EEG signals.

5.3.2. SET-BASED INTEGER-CODED FUZZY GRANULAR EVOLUTIONARY ALGORITHMS FOR CHANNEL SELECTION:

SIFE [55] is an evolutionary algorithm that combines concepts from set theory with fuzzy granulation wrapper techniques. By integrating fuzzy granulation, it simplifies how the population is initialized. This approach helps creation of a varied population over multiple iterations and reduces the need for repeated fitness evaluations by acting as an alternative evaluation method.

Feature selection methodologies focus on finding $s* \subseteq \Omega$, where $s* = f1*$, $f2*, \ldots, fK*$ represents the most optimal and concise subset from the possible feature set $\Omega = f1, f2, \ldots, fnF$. In this situation, nF indicates the total number of features, while K defines the optimal feature count. Algorithm 5.2 provides SIFE implementation pseudocode.

Algorithm 5.2: SIFE Channel selection

$Set\ N, vt\ , nC, nM, n\mu, \Omega, t\ \leftarrow\ 0, tmax$
$P \leftarrow Repulsive\ fuzzy\ granular\ initialization\ (N,\ v_t)$
$Pool\ \leftarrow\ P$
$While\ termination\ criteria\ not\ met\ do$
$\quad t \leftarrow t + 1$
$\quad select\ chromosomes\ by\ Roulette - wheel$
$\quad P_C\ \leftarrow\ call\ n_c\ times\ the\ three - parent\ UI\_Crossover$
$\quad select\ chromosomes\ randomly$
$\quad P_M\ \leftarrow\ call\ n_m\ times\ the\ C\_Mutation$
$\quad Offspring\ \leftarrow\ P_C\ \cup\ P_M$
$\quad for\ 1\ to\ |\ offspring\ |\ do$
$\quad\quad s_i\ \leftarrow\ retrieve\ the\ i^{th}\ solution\ from\ the\ offspring$
$\quad\quad M_s\ \leftarrow\ Fuzzy\ similarity\ (Pool, s_i)$
$\quad\quad if\ M_s\ >\ v_t$
$\quad\quad\quad \mathcal{F}(s_i)\ \leftarrow\ retrieve\ its\ fitness\ from\ the\ pool$
$\quad\quad else$
$\quad\quad\quad \mathcal{F}(s_i)\ \leftarrow\ calculate\ the\ fitness$
$\quad\quad\quad\quad Pool\ \leftarrow\ Pool\ \cup\ \{\ s_i\ ;\ \mathcal{F}(s_i\ )\}$
$\quad\quad end\ if$
$\quad end\ for$
$\quad sort\ the\ pool\ based\ on\ the\ fitness$
$\quad control\ the\ pool\ size\ and\ update\ \ v_t\ by\ (3)$
$\quad P(t)\ \leftarrow\ restore\ the\ N\ first\ granule\ from\ the\ pool$
$\quad end\ while$
$return\ the\ first\ solution\ in\ Pt_{max}\ as\ optimal\ subset\ s\ *$

Algorithm 5.3 outlines the initial step for creating the repulsive fuzzy granular population by iteratively generating a random individual *si* within the search space. This process continues until a total of *N* individuals form the starting population *P*. The commonly used "roulette-wheel" method is applied to maintain accurate issue representation using integer coding. Recombination or crossover, is crucial for effective convergence. The total amount of request made to the three-parent UI crossover is $n_c = P_C * \frac{N}{2}$, where the crossover probability probability is P_C, N is the population size. It establishes the size of the population of offspring. To ensure population diversity and avoid settling into local optima, a complement mutation (C-Mutation) is used. The complement feature set $\bar{si}$, which includes all the attributes from the universal set (Ω) that are not present in the currently chosen individual s_i, is used in this mutation to induce randomization. A new offspring is created by randomly selecting $n\mu$ features (genes) from $\bar{si}$ and replacing them with $n\mu$ features from s_i. A new produced offspring is given as by $n_m = p_M * N$, where p_M is the mutation probability. A collection of fuzzy granules is subsequently created by fuzzy granulation—representations of people who have

previously been evaluated. These granules compete, according to a similarity metric, to absorb and evaluate newly created solution s_i.

Algorithm 5.3: Repulsive Fuzzy Granular Initialization (N, ϑ)

---

1. $i \leftarrow 1, P \leftarrow$
   $create\ a\ solution\ randomly\ within\ the\ search\ space\ and\ evaluate\ it.$
2. $While\ i \leq N\ do$

   $s_i \leftarrow create\ a\ solution\ randomly\ within\ the\ search\ space$
   $M_s \leftarrow FuzzySimilarity(P, s_i)\ (see\ Algorithm\ 7.5)$
   $If\ M_s < \vartheta\ then\ \longrightarrow\ there\ is\ no\ significant\ resemblance$
   $\quad F(s_i) \leftarrow evaluate\ the\ s_i$
   $\quad P \leftarrow P \cup \{s_i, F(s_i)\}, i \leftarrow i + 1$
   $End\ if$

3. $End\ while$
4. $Return\ P\ as\ the\ initial\ population$

Algorithm 5.3 describes how to initialize the repulsive fuzzy granular population by repeatedly generating random individuals in the search space. A fuzzy similarity metric is used to assess each individual, and only those that differ from all previously created individuals are added to the original population. This process continues until the initial population P reaches the desired size of N individuals.

Algorithm 5.4: Three-parent UI_Crossover (s_1, s_2, s_3)

---

1. $O_1 \leftarrow s_1 \cap s_2 \quad \longrightarrow intersection\ operation$
2. $J_1 \leftarrow s_1 \cap s_3$
3. $J_2 \leftarrow s_2 \cap s_3$
4. $O_2 \leftarrow O_1 \cup J_1 \cup J_2 \quad \longrightarrow union\ operation$
5. $If\ O_1 = \emptyset\ then\ \longrightarrow\ O_1 \leftarrow create\ a\ solution\ randomly$
6. $If\ O_2 = \emptyset\ then\ \longrightarrow\ O_2 \leftarrow create\ a\ solution\ randomly$
7. $Return\ O_1, O_2\ as\ two\ offspring$

Crossover, also known as recombination, is essential for ensuring the algorithm converges accurately and for leveraging the diversity in the solution's population. In this case, a new three-parent UI crossover method was introduced, which combines the set theory concepts of 'union' and 'intersection,' as described in Algorithm 5.4. Three parents are used—two selected using the roulette wheel method and one randomly picked to maintain diversity. Using these, 2 new offspring were developed by employing the intersection and union operations. These offspring inherit

the most desirable traits from their parents. The intersection operation keeps only shared genes, helping to minimize the number of features, following a greedy or contraction-based approach. On the other hand, the union operation retains all valuable attributes while eliminating redundancy, thus enabling a broader search. This crossover method avoids random elements that could mislead the search process. By relying on union and intersection, it minimizes the chances of destroying good solutions and provides more information transfer than traditional crossover techniques. By using three parents to create more productive offspring, it also shares more information than traditional crossover techniques. The created offspring population size is determined by the crossover probability P_C, and the total amount of request made to the three-parent UI crossover is $n_c = P_C * \frac{N}{2}$, where N is the population size.

Algorithm 5.5: $C\_Mutation(s_i, \Omega, n_u)$

---

1. $\bar{s}_i \leftarrow \Omega - s_i \quad \longrightarrow complement\ operation$
2. $indx \leftarrow select\ randomly\ n_u\ indexes\ from\ \bar{s}_i$
3. $idM \leftarrow select\ randomly\ n_u\ genes\ from\ s_i$
4. $s_i(idM) \leftarrow \bar{s}_i(indx)$
5. $Return\ s_i\ as\ the\ mutated\ solution$

Algorithm 5.6: $FuzzySimilarity(pool, s_i)$

---

1. $For\ j = 1\ to\ |pool|\ do$
2. $\quad s_j \leftarrow retrieve\ position\ of\ j^{th}\ granule\ from\ the\ pool$
3. $\quad \mu_j \leftarrow exp(-(s_i - s_j)^2) \quad \longrightarrow similarity\ measure$
4. $\quad \bar{\mu}_j \leftarrow average\ of\ \mu_j\ over\ decision\ variables$
5. $End\ for$
6. $M_s \leftarrow max(\bar{\mu}_j)$
7. $Return\ M_s\ as\ maximum\ similarity\ between\ the\ pool\ and\ s_i$

The Mutation is a vital component in evolutionary algorithms, helping the system escape local optima during the search process. In this work, we introduce a complement mutation (C-Mutation) operator that enhances population diversity and avoids getting stuck in suboptimal solutions, as detailed in Algorithm 5.5. This operator increases unpredictability by using a complement feature set s_i, which includes every features from the universal set (Ω) that are missing in the currently chosen individual s_i.

One major obstacle in using metaheuristic algorithms, particularly for large-scale optimization problems, is the high computational cost. While

efficient coding techniques and variation operators can help, these methods still face significant limitations in very high-dimensional (VHD) scenarios. To address this, we apply fuzzy granulation (FG) to reduce the computational load during the search. FG operates by linking an individual's fitness to how similar it is to another person whose level of fitness is already recognized. It develops a database of fuzzy granules, which are representations of solutions that have already been assessed. As detailed in Algorithm 5.6, if a new solution is generated, these granules in the pool compete to integrate the newcomer based on a fuzzy similarity score. The similarity between s_i and s_j (s_j is j^{th} granule) is calculated as:

$$\mu_j = exp\left(-\left(s_i - s_j\right)^2\right) \tag{5.17}$$

The average similarity or mean $\bar{\mu j}$ is computed across all the variables. The extreme similarity between the pool and si is defined as $M_s = \max(\bar{\mu j})$, computed as:

$$\mathcal{F}(s_i) = \begin{cases} F_s & if\ M_s > v_t \\ \mathcal{F}(s_i) & if\ M_s \le v_t \end{cases} \tag{5.18}$$

If this similarity M_s is below a predefined threshold v_t, the new solution is evaluated and added to the pool, initializing its count at 1. Otherwise, the existing granule (F_s) inherits the new solution's fitness value. The count mechanism filters out granules that no longer correspond to current solutions, thus avoiding excessive memory usage and unnecessary computation. Adaptive threshold v_t is given by:

$$v_t = \rho * mean\ F(P(t-1))/max\ F(P(t-1)) \tag{5.19}$$

Above threshold is governed by a balancing parameter ρ, which ranges between 0 and 1 to maintain an effective balance between algorithm's central performance and iteration efficiency. The population at any generation t is denoted as P (t). FG relies on the principle that similar solutions yield similar results, a concept it leverages to skip redundant evaluations and boost efficiency—particularly if this opinion holds true in a provided optimization scenario.

5.4. TRADITIONALS CLASSIFIERS:

Classification is a fundamental task in ML that includes sharing data into predefined classes based on feature of input. Traditional machine learning algorithms like SVM, KNN, and RF have been widely used for their efficiency in pattern recognition tasks including as signal categorization and medical diagnosis. These models learn from labelled training data to distinguish between different classes with high accuracy. Ensemble learning methods, such as Ada-Boost, XG-Boost, and bagging-based techniques, combine the classification of different base classifiers to

enhance overall efficiency. In this study, task classification was performed using SVM, KNN, Ada-Boost, XG-Boost, and Random Forest.

5.4.1. Support Vector Machine (SVM):

SVM [56] is a robust supervised ML method that is mostly utilized for categorization problems, while it may also be used for regression problems. SVM works by finding the best hyperplane which is divides the data into multiple groups. In two-dimensional space, this hyperplane is only a line. In higher dimensions, it turns into a plane or manifold. The main goal is to make the margin as big as possible. The created margin is the space between the nearby data points of each class and the hyperplane, which are termed support vectors. A bigger margin means that the model can better handle new data. SVM employs kernel functions (such polynomial [57], RBF [58], or sigmoid [59]) to move the feature into a higher dimensional space that separates hyperplane which can be identified for data that can't be separated by a straight line.

Advantages of SVM:

- Even when there are more dimensions than samples, it is still efficient in high dimensional settings.

- Effective if that is a distinct gap between classes.

- Memory-efficient since it makes decisions based solely on support vectors.

- Versatile, as different kernels can be applied depending on the data nature.

Disadvantages of SVM:

- Not suitable for large datasets, because it requires high training time.

- Less effective when the data is very noisy or overlapping.

- Choosing the right kernel and tuning parameters like C and gamma can be complex and computationally expensive.

- Difficult to interpret, especially when using non-linear kernels.

5.4.2. K-Nearest Neighbours (KNN):

KNN is a simple instance-based supervised ML technique which can be employed for classification and regression [60]. It operates on the tenet that comparable data points can be found nearby in feature space. Working of KNN discussed below:

- KNN examines the "K" nearest neighbours (data features) in the dataset of training when a new data point needs to be classified

- It then allocates the most mutual class (in categorization) or computes the mean value (in regression) among those K neighbours.

- Distance is usually calculated using Euclidean distance, though other metrics like Manhattan or Minkowski distance can also be used.

Advantages of KNN:

- Easy to comprehend and apply.

- It is a lazy learner because there is no training phase.

- Adaptable to several-class problems.

- Naturally handles non-linear decision boundaries.

Disadvantages of KNN:

- Slow prediction time for large datasets (since it calculates distance to all training samples).

- Memory intensive, as it stores the entire dataset.

- Sensitive to irrelevant features and the scale of the data.

- Choosing the right K value is critical and can impact performance.

5.4.3. Random Forest:

An ensemble ML approach namely Random Forest [61] is employed to both classification and regression problems. During training, it builds a number of decision trees and outcomes either the majority vote (for classification) or the average prediction (for regression). Working of random forest discussed below:

- **Bootstrap Sampling (Bagging):** It generates a number of bootstrap samples, which are random portions of the early training dataset with replacement. A different decision tree is trained for each subset.

- **Feature Randomness:** To determine the optimal split, a decision tree chooses a random portion of features at each node rather than taking into account every feature. This increases diversity among the trees and reduces correlation and improving overall performance.

- **Tree Building:** Each decision tree is grownup independently and to the maximum depth (usually unpruned). Trees are trained on different data and features.

- **Prediction:** Random Forest predicts the class based on majority voting across all trees.

Advantages of Random Forest:

- **High Accuracy:** Combines multiple decision trees for better prediction performance.

- **Robust to Overfitting:** Randomization and averaging reduce overfitting compared to a single tree.

- **Handles High Dimensionality:** Works well even with large numbers of features.

- **Feature Importance:** Provides understandings into which features are extremely influential in prediction.

- **Handles Missing Values & Outliers:** Performs well even with imperfect data.

Disadvantages of Random Forest:

- **Less Interpretability:** While individual decision trees are easy to interpret, a forest of many trees is not.

- **Computational Cost:** Training many trees can be time and resource-intensive.

- **Slower Predictions:** Especially with many deep trees, prediction can be slower than simpler models.

- **Storage Complexity:** Requires more memory due to multiple trees being stored.

5.5.   ENSEMBLE CLASSIFIERS:

Ensemble learning [62] methods, such as Ada-Boost, XG-Boost, and bagging-based techniques, combine the classifications of multiple base classifiers to enhance overall performance [63]. These approaches enhance robustness and accuracy by reducing overfitting and leveraging the strengths of various learners. In EEG signal analysis or biomedical applications, combining machine learning with ensemble methods often leads to more reliable and generalizable classification results. In this study, task classification was performed using SVM, KNN, Ada-Boost, XG-Boost, and Random Forest.

5.5.1. Ada-Boost (Adaptive Boosting):
An ensemble learning method called Ada-Boost [64] develops a powerful classifier by merging several weak learners, usually decision trees. It trains weak learners iteratively, with each new model focusing more on the cases that the earlier models misclassified. The main concept is to give misclassified samples more weights in order to encourage future classifiers to focus more on these more challenging cases. Working of adaboost discussed below:

- **Initialization:** All training samples are initially assigned equal weights.

- **Sequential Learning:** Using the dataset, a weak learner—such as a decision stump—is trained. This weak learner's mistake rate is computed.

The next learner concentrates more on challenging cases by increasing the weights of misclassified samples. This procedure is carried out repeatedly until performance stabilizes or for a certain number of rounds.

- **Final Prediction:** The accuracy of each weak learner's output is used to weight it. A weighted aggregate (for regression) or weighted majority vote (classification) is the ultimate forecast.

Advantages of Ada-Boost:

- **Improved Accuracy:** Converts weak learners into a strong ensemble with better performance.

- **Simple and Versatile:** Can be used with various base learners (though decision stumps are common).

- **Focuses on Difficult Cases:** Learns from errors by emphasizing misclassified samples.

- **Less Overfitting:** Often generalizes well when not over-trained.

Disadvantages of Ada-Boost:

- **Sensitive to Noisy Data:** Focuses on hard samples, which can include outliers or mislabelled data.

- **Requires Careful Tuning:** Learning rate and the number of weak learners affect model efficiency.

- **Not Parallelizable:** Since each and every learner depends on the previous one, training can't be parallelized easily.

- **Less Interpretability:** As with many ensemble methods, difficult to understand the model's decision path.

5.5.2. XG-Boost (Extreme Gradient Boosting):
XG-Boost [41] [65] is an advanced implementation of gradient boosting that is highly efficient and scalable. It builds an ensemble of decision trees by optimizing a specific loss function using gradient descent. XG-Boost introduces regularization parameters to reduce overfitting and improves speed with parallel processing. Working of Ada-boost discussed below:

- **Boosting Framework:** XG-Boost successively constructs an ensemble of decision trees, just like Ada-Boost. By minimizing a loss function, each new tree is trained to correct the errors (residuals) of the previous ensemble.

- **Gradient Descent:** The loss function is minimized by the model using gradient descent. Instead of updating weights like Ada-Boost, XG-Boost uses gradients (first and second-order) to optimize the model.

- **Regularization:** Includes L1 and L2 regularization to avoid overfitting, which makes it more robust compared to traditional boosting.

- **Tree Pruning:** Uses max depth and leaf-wise tree growth, which makes it faster and more efficient.

- **Handling Missing Data:** It can instantly acquire the ability to deal with missing values during training.

Advantages of XG-Boost:

- **High Accuracy:** Often achieves state-of-the-art performance.

- **Speed:** Optimized for parallel computation and fast execution.

- **Regularization:** Reduces overfitting through L1/L2 penalties.

- **Flexible:** Works with different loss functions and supports classification, regression, and ranking.

- **Robust:** Handles missing values well and supports early stopping.

Disadvantages of XG-Boost:

- **Complexity:** More difficult to understand and tune compared to simpler models like KNN or SVM.

- **Requires Parameter Tuning:** Needs careful tuning of hyper-parameters for optimal performance.

CHAPTER 6

RESULTS AND DISCUSSION

6.1. DATA VISUALIZATION:

The process is implemented with the BCI - IV - 2A dataset using python coding in the Google Collab Notebook. In this work, there are few steps which we have to follow during the data preprocessing. The raw EEG data sample and preprocessed signal shown in Figure 6.1 and 6.2 respectively.
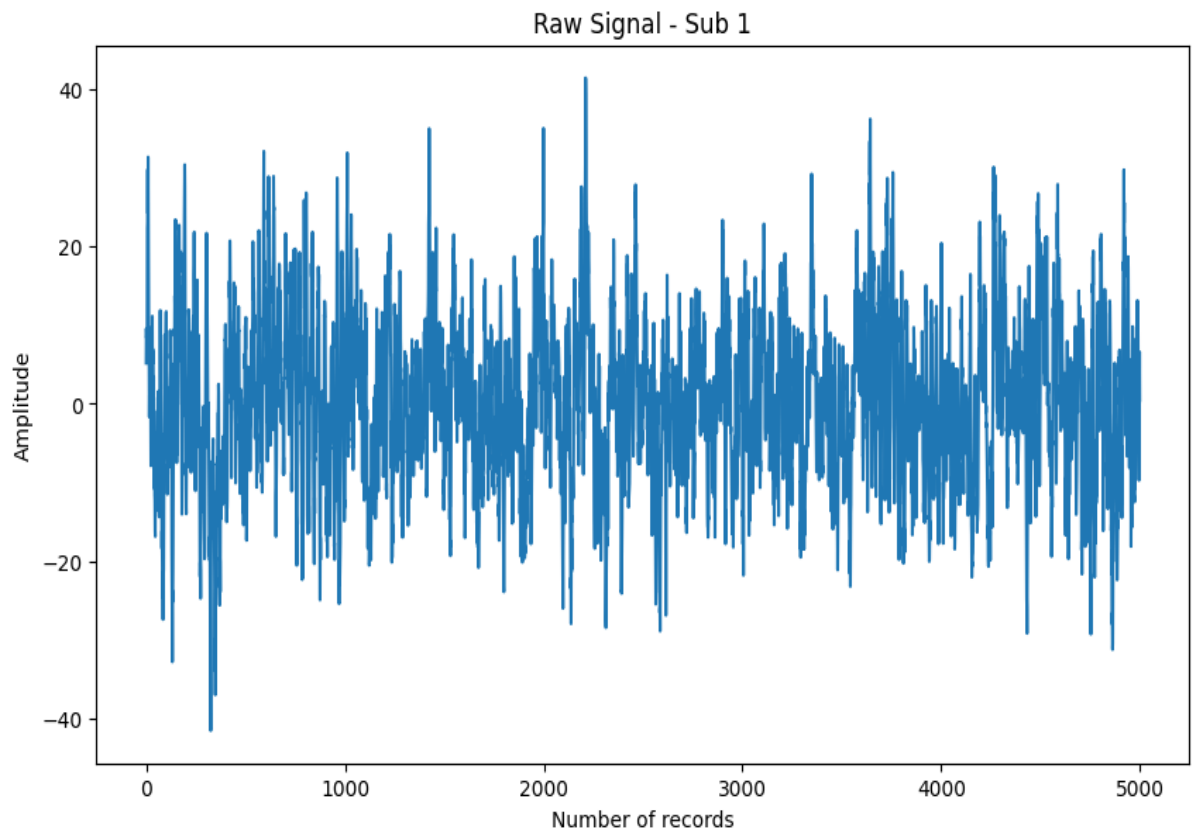

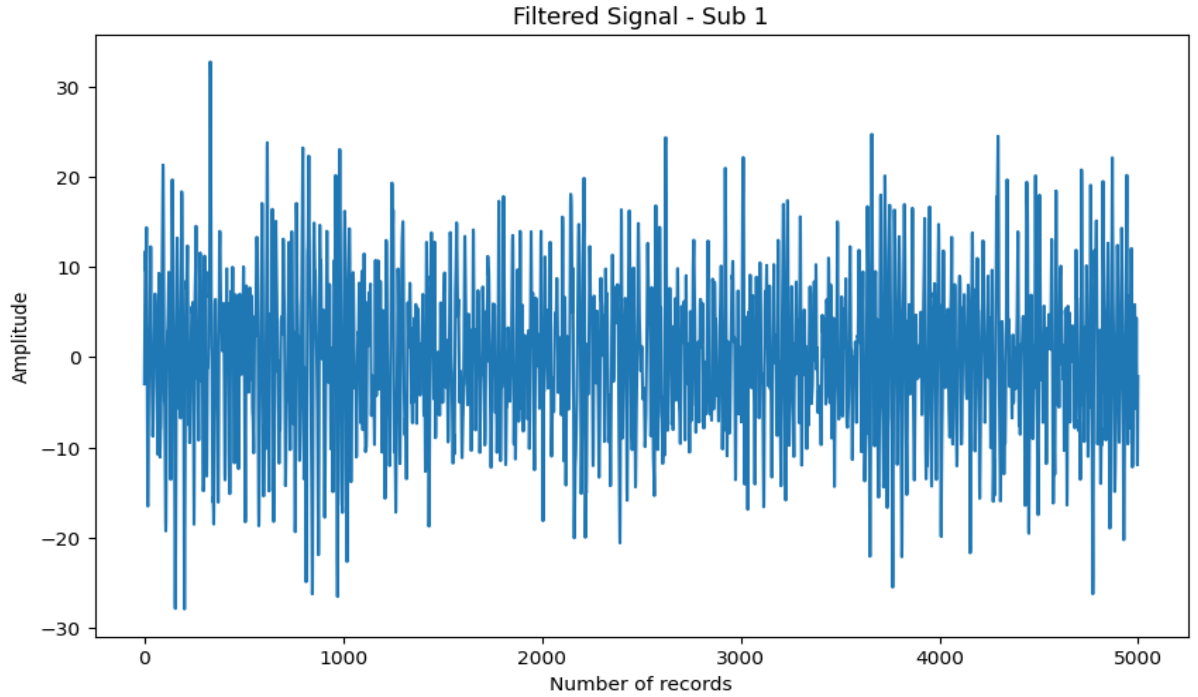**Figure 6.1. Sample of Raw EEG signal**

Figure 6.2. After preprocessing (4 - 40Hz signal)

6.2.   PERFORMANCE METRICS:

The accuracy, recall, precision, F1 score, AUC curve, and confusion matrix were among the metrics utilized in this study to evaluate the classification models' efficiency:

6.2.1. Accuracy:

Accuracy is defined as the percentage of accurate classifications, whether negative or positive.

$$Accuracy = \frac{TP+TN}{TP+ TN+ FP+ FN} \qquad (6.1)$$

Where TPs: True Positives, FPs: False Positives, TNs: True Negatives and FNs: False Negatives.

6.2.2. Precision:

Shows the percentage of TP predictions out of all positive predictions, evaluating how well the model finds pertinent examples.

$$Precision = \frac{TP}{TP+ FP} \qquad (6.2)$$

6.2.3. Recall (Sensitivity):

Calculates the ratio of TPs to the sum of TPs and TNs, which evaluates the model's accuracy in identifying all pertinent events

$$Recall = \frac{TP}{TP+ TN} \qquad (6.3)$$

6.2.4. F1 Score:

F1 score is a harmonic mean of precision and recall that measure a balancing FPs and FNs.

$$F1\ score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \qquad (6.4)$$

6.2.5. Kappa score:

A statistical metric known as the Kappa Score assesses the degree of agreement between genuine labels and anticipated classifications while also taking chance agreement into consideration. It is especially helpful for datasets that are unbalanced where simple accuracy might be misleading. Generally, the Kappa score ranges between -1 to 1:

$$Kappa = 2 \times \frac{p_a - p_e}{1 - p_e} \qquad (6.5)$$

Where p_a observed agreement (actual accuracy), and p_e expected agreement by chance.

6.2.6. Area Under the Curve (AUC):

The capacity of a model to differentiate among classes is measured by its AUC. It is based on the ROC curve and has a range between 0 to 1, with 0.5 representing random prediction and 1 representing accurate classification. Better overall performance is indicated by a higher AUC score, which is particularly helpful for datasets that are unbalanced.

6.2.7. Confusion Matrix:

A matrix indicating the total number of true positives (TPs), true negatives (TNs), false positives (FPs), and false negatives (FNs) provides an overview of the classification model's performance, offering a detailed insight into prediction errors.

6.3. RESULTS AND DISCUSSION FOR FBCSP-PSO & AGSP-SIFE CHANNEL SELECTION USING MACHINE LEARNING:

Table 6.1 presents the selected EEG channels for each subject using two distinct channel selection approaches: FBCSP-PSO and AGSP-SIFE. The table highlights how each method identifies subject-specific subsets of EEG channels that are most informative for MI categorization. The FBCSP-PSO method typically selects a larger set of channels, often including both central and peripheral regions, suggesting that it captures a broader spatial representation of motor-related brain activity. For example, Subject 2 has 16 channels selected under FBCSP-PSO, indicating its tendency to retain diverse spatial features that may contribute to improved classification accuracy. In contrast, the AGSP-SIFE method generally selects a more refined and minimal subset of channels while maintaining performance. This is evident in subjects like S3 and S8, where AGSP-SIFE significantly reduces the number of selected channels compared to FBCSP-PSO.

Table 6.1: Selected channel list of FBCSP-PSO & AGSP-SIFE channel selection method

Sub	FBCSP-PSO	AGSP-SIFE
1	[0,1,2,5,6,7,8,9,12,14,15,17,18,20,21]	[5 , 6, 7, 9, 10 ,15, 16 ,18, 20 ,21]
2	[0,3,4,5,6,7,9,10,11,13,14,15,16,17,19,21]	[0 , 3 ,7 , 9, 10, 13, 14, 17, 18 ,20, 21]
3	[0,2,3,7,8,10,13,14,15,17,20,21]	[2 , 4 , 6 , 8 , 9 ,10 ,12]
4	[0,1,2,5,6,7,8,10,11,12,14,17,18,19,20]	[0, 1 , 2 ,6, 7, 11 ,14, 16, 20, 21]
5	[1,2,4,7,8,17,18,21]	[0 ,1 , 2 , 5 , 8, 10 ,12, 13, 14 ,18, 19, 21]
6	[0,1,2,3,4,5,9,10,12,13,15,16,17,18,21]	[1 , 2 , 4 , 6 , 9 ,12, 13, 17 ,18, 19]
7	[2,3,4,6,7,8,9,11,12,14,15,16,17,18,20]	[0, 1, 2, 3, 5, 6 , 8 , 9 ,10, 11, 13, 14 ,15 ,16]
8	[1,3,4,7,8,9,10,12,13,14,16,17,19,21]	[1 , 6 ,9 ,10, 14, 17 ,18]
9	[0,2,3,6,7,8,9,10,11,12,13,21]	[0, 1, 2, 3 , 4, 5, 8, 10, 17 ,18, 19]

This reduction reflects AGSP-SIFE's focus on maximizing discriminative power through advanced signal processing and spectral graph theory, resulting in compact yet effective channel sets. Channels like 10, 14, 17, and 21 are consistently selected across many subjects in both methods, underlining their significance in motor imagery tasks. These frequently selected channels are located near the sensorimotor cortex, which is critical for MI signal detection. Overall, this comparison demonstrates that while FBCSP-PSO offers a comprehensive channel selection, AGSP-SIFE enhances efficiency by selecting fewer, more targeted channels—making it highly appropriate for real-time and resource-constrained BCI related applications without compromising classification accuracy.

Table 6.2: Performance analysis of FBCSP-PSO & AGSP-SIFE channel selection method using Machine learning classifiers on for all 9 subjects

Sub	Metrics	FBCSP-PSO SVM	FBCSP-PSO RF	FBCSP-PSO KNN	AGSP-SIFE-RF	AGSP-SIFE-XGB	AGSP-SIFE-ADA
S1	Accuracy	90.85	86.62	80.99	90.14	99.3	99.3
	Precision	97.1	87.34	94.74	96.72	98.31	98.61
	Recall	85.9	88.46	69.23	83.1	100	100
	F1 Score	91.16	87.9	80	89.39	99.3	99.3
	Kappa	0.82	0.73	0.63	0.8	0.99	0.99
	AUC	0.91	0.86	0.82	0.9	0.99	0.99
S2	Accuracy	80.99	74.65	83.1	92.96	99.3	99.3
	Precision	75.64	76.27	79.45	96.92	98.61	98.61

	Recall	88.06	67.16	86.57	88.73	100	100
	F1 Score	81.38	71.43	82.86	92.65	99.3	99.3
	Kappa	0.62	0.49	0.66	0.86	0.99	0.99
	AUC	0.81	0.74	0.83	0.93	0.99	0.99
S3	Accuracy	92.25	88.73	86.62	97.89	99.3	98.59
	Precision	94.52	89.47	90.14	100	100	100
	Recall	90.79	89.47	84.21	95.77	98.59	97.18
	F1 Score	92.62	89.47	87.07	97.84	99.29	98.57
	Kappa	0.84	0.77	0.73	0.96	0.99	0.97
	AUC	0.92	0.89	0.87	0.98	0.99	0.99
S4	Accuracy	90.14	83.8	83.8	91.55	90.85	95.07
	Precision	95.31	85.51	87.69	91.55	100	92.11
	Recall	84.72	81.94	79.17	91.55	81.69	98.59
	F1 Score	89.71	83.69	83.21	91.55	89.92	95.24
	Kappa	0.8	0.68	0.68	0.83	0.82	0.9
	AUC	0.9	0.84	0.84	0.92	0.91	0.95
S5	Accuracy	93.66	92.25	92.965	94.37	94.37	95.77
	Precision	95.59	94.12	92.96	100	100	100
	Recall	91.55	90.14	92.963	88.73	88.73	91.55
	F1 Score	93.55	92.09	92.963	94.03	94.03	95.59
	Kappa	0.87	0.85	0.86	0.89	0.89	0.92
	AUC	0.94	0.92	0.93	0.94	0.94	0.96
S6	Accuracy	88.57	80	85.71	81.88	96.38	94.93
	Precision	85.53	79.17	82.05	78.21	100	100
	Recall	92.64	81.43	91.43	88.41	92.75	89.86
	F1 Score	89.04	80.28	86.49	82.99	96.24	94.66
	Kappa	0.77	0.6	0.71	0.64	0.93	0.9
	AUC	0.89	0.8	0.86	0.82	0.96	0.95
S7	Accuracy	88.03	85.21	85.21	81.69	86.62	86.62
	Precision	81.94	76.25	77.63	80.82	85.14	91.94
	Recall	93.65	96.836	93.65	83.1	88.73	80.28
	F1 Score	87.41	85.31	84.89	81.94	86.9	85.71
	Kappa	0.76	0.71	0.71	0.63	0.73	0.73
	AUC	0.89	0.86	0.86	0.82	0.87	0.87
S8	Accuracy	94.37	83.8	85.92	96.48	94.37	82.39
	Precision	94.03	79.73	85.07	100	94.37	91.07
	Recall	94.03	88.06	85.07	92.96	94.37	71.83
	F1 Score	94.03	83.69	85.07	96.35	94.37	80.31
	Kappa	0.89	0.68	0.72	0.93	0.89	0.65
	AUC	0.94	0.84	0.86	0.96	0.94	0.82
S9	Accuracy	81.69	77.46	78.17	86.62	92.25	93.66
	Precision	77.03	73.61	75.36	84.21	87.5	91.89
	Recall	86.36	80.3	78.79	90.14	98.59	95.77
	F1 Score	81.43	76.81	77.04	87.07	92.72	93.79
	Kappa	0.63	0.55	0.56	0.73	0.85	0.87

AUC	0.82	0.78	0.78	0.87	0.92	0.94
Mean accuracy	**88.95**	**83.61**	**84.72**	**90.39**	**94.74**	**95.95**

Table 6.2 presents a comprehensive performance evaluation of two channel selection methods—FBCSP-PSO and AGSP-SIFE—across nine subjects using various ML classifiers such as SVM, K-Nearest Neighbors (KNN), Random Forest (RF), XG-Boost (XGB), and Ada-Boost (ADA). The analysis covers several performance metrics including Accuracy, Cohen's Kappa, Recall, Precision, F1 Score, and AUC. The outcomes represents that the AGSP-SIFE technique consistently beats the traditional FBCSP-PSO technique across all classifiers and subjects. Notably, AGSP-SIFE combined with ensemble classifiers (XGB and ADA) achieves the highest accuracy for most subjects, with mean accuracies of 94.74% and 95.95% respectively. In contrast, the FBCSP-PSO method records a comparatively lower average accuracy, with 88.95% using SVM, 83.61% using RF, and 84.72% using KNN.
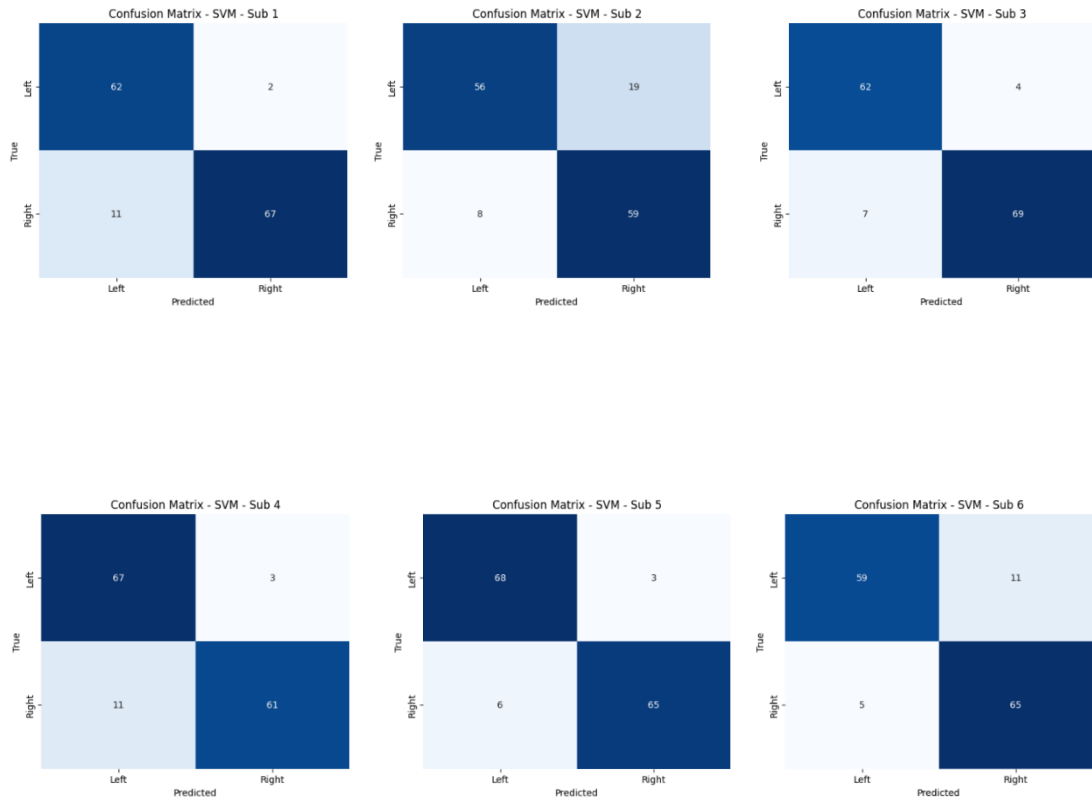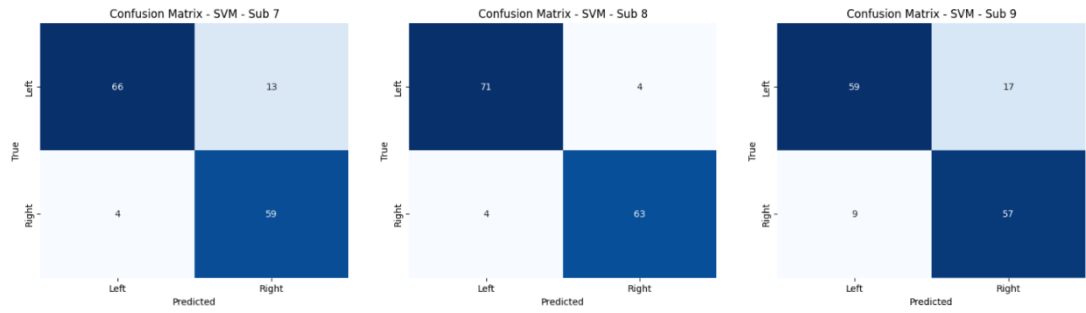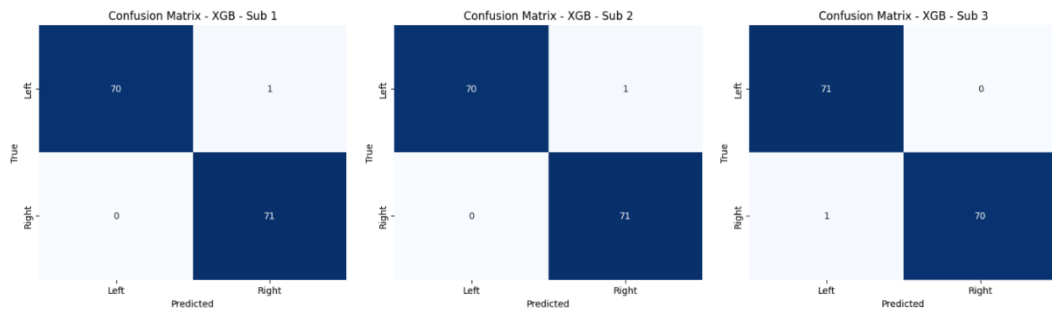
Figure 6.3. Confusion matrices of FBCSP-PSO-SVM for all 9 subjects

Subject-wise investigation illustrations that AGSP-SIFE-ADA accomplishes the uppermost performance for several subjects (Sub 1, Sub 2, Sub 3, Sub 5, and Sub 9) in terms of both F1 score and accuracy, reaching 99.3% accuracy for S1 and S2. Even in more challenging cases like S6 and S7, AGSP-SIFE maintains robust performance, outperforming FBCSP-PSO in terms of AUC and Kappa values. The superiority of AGSP-SIFE can be attributed to its ability to extract spatially informative features more effectively, which enhances the discriminative power of the classifiers.

Among the classifiers, Ada-Boost and XG-Boost, when paired with AGSP-SIFE, demonstrate better generalization across subjects due to their ensemble nature, which reduces bias and variance. On the other hand, FBCSP-PSO shows reasonably good performance with SVM, particularly in subjects like S3, S4, and S5, but fails to maintain consistent performance across the entire subject pool. In summary, the experimental outcomes strongly recommend that the AGSP-SIFE technique, especially when used with Ada-Boost or XG-Boost, is a highly effective method for channel selection and classification in EEG-related tasks. Using this technique not only improves classification performance but also enhances robustness, as evidenced by high Kappa and AUC values across all subjects.
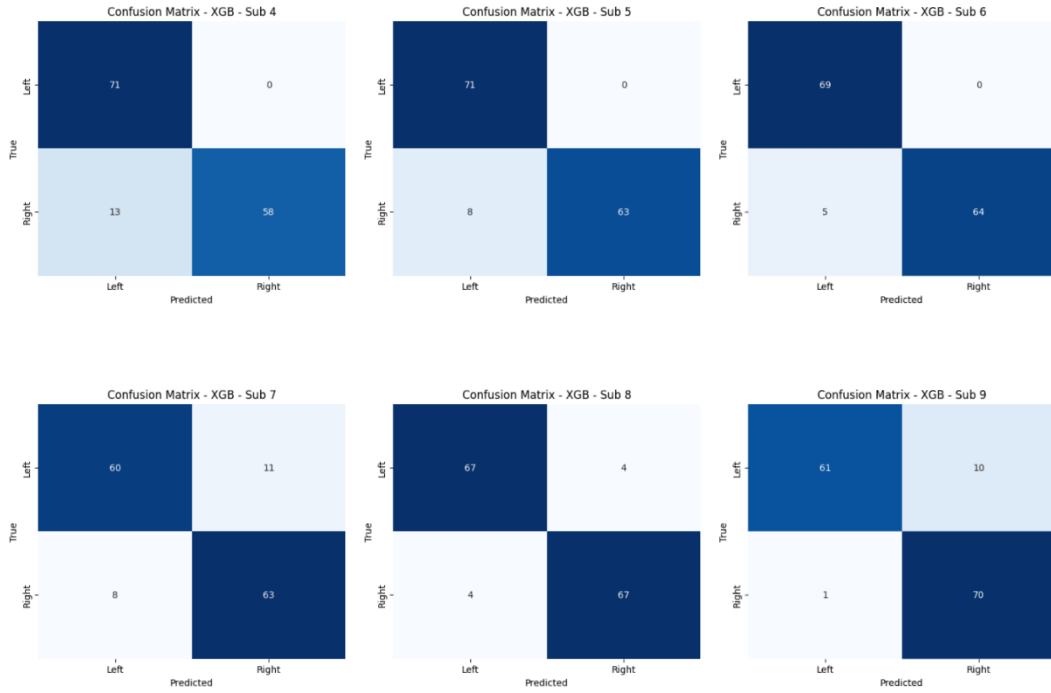
Figure 6.4. Confusion matrices of AGSP-SIFE-XGB for all 9 subjects

Figure 6.3 and 6.4 shows confusion matrix of FBCSP-PSO and AGSP-SIFE channel selection algorithms. The confusion matrices for FBCSP-PSO and AGSP-SIFE methods reveal insightful details about each model's classification behavior across the nine subjects. Each confusion matrix is representing the count of true positives, false negatives, false positives, and true negatives respectively. This allows us to assess the balance between recall and specificity. For Subject 1, FBCSP-PSO misclassified 2 positive and 11 negative samples, resulting in slightly lower performance, whereas AGSP-SIFE nearly perfectly classified the samples with only one false negative and zero false positives, demonstrating very high recall and specificity.

In Subjects 2 and 3, FBCSP-PSO showed a higher number of false negatives (19 and 4 respectively) and some false positives (8 and 7), suggesting some imbalance in detecting both classes. However, AGSP-SIFE again demonstrated a near-perfect classification, with only one false negative and no false positives, highlighting its robustness and reliability. A slight exception appears in Subject 4, where AGSP-SIFE had 13 false positives compared to FBCSP-PSO's 11, indicating a trade-off in precision despite having more true positives (71 vs. 67). This pattern is also seen in Subject 5, where both methods performed well, but AGSP-SIFE still showed slightly better classification for the positive class. For Subject 6, AGSP-SIFE reduced both false negatives and false positives compared to FBCSP-PSO, resulting in a more balanced confusion matrix. Similarly, for Subject 7, AGSP-SIFE still misclassified more negatives than FBCSP-PSO (11 vs. 13), but made fewer false positives (8 vs. 4), showing better positive class detection. In Subject 8, both methods exhibited nearly

identical classification performance, with AGSP-SIFE providing slightly more balanced predictions (equal false positives and negatives) compared to FBCSP-PSO, which favored the positive class more. Finally, Subject 9 shows one of the largest improvements from FBCSP-PSO to AGSP-SIFE. FBCSP-PSO had 17 false negatives and 9 false positives, significantly lowering its F1 score. In contrast, AGSP-SIFE misclassified only one positive and 10 negatives, clearly showing superior detection of the target class. Overall, the confusion matrices confirm that AGSP-SIFE consistently improves true positive rates and significantly reduces false classifications compared to FBCSP-PSO. This directly supports the performance gains observed in classification metrics such as accuracy, F1 score, and AUC, making AGSP-SIFE a more reliable and effective channel selection approach for EEG signal classification tasks.

6.4. PERFORMANCE ANALAYSIS OF OPTIMIZATION-BASED VARIOUS CHANNEL SELECTION TECHNIQUES:

To estimate the efficiency of the proposed FBCSP-PSO-SVM and AGSP-SIFE-XGB methods, we conducted a comparative examination with several existing state-of-the-art methods: MX-BBOA-SVM, LS-BJOA-SVM, FCNNA, and MI-BMInet across nine subjects. Table 6.3 represents the performance analysis of FBCSP-PSO - SVM, AGSP-SIFE-XGB with various existing channel selection method using Machine learning classifiers on for all 9 subjects. The evaluation metric used is classification accuracy (%), and the outcomes obviously illustrate the dominance of our proposed methods, especially AGSP-SIFE-XGB. Figure 6.5 shows accuracy comparison of many channel selection algorithms. Among all methods, AGSP-SIFE-XGB consistently achieved the highest accuracy, scoring above 99% in subjects 1 to 3, and maintaining strong performance across all other subjects. Its average accuracy is 94.75%, which is significantly higher than the others.

Table 6.3: Performance analysis of various channel selection method using Machine learning classifiers on for all 9 subjects in term of accuracy

Sub	FBCSP-PSO-SVM	AGSP-SIFE-XGB	MX-BBOA-SVM	LS-BJOA-SVM	FCNNA	MI-BMInet
1	90.85	**99.3**	84.33	78.42	88.65	86.98
2	80.99	**99.3**	78.2	78.42	88.03	72.65
3	92.25	**99.3**	91.33	83.33	98.54	94.95
4	90.14	**90.85**	83.02	88.46	90.52	76.66
5	93.66	94.37	87.66	91.89	**97.78**	93.84
6	88.57	**96.38**	80.04	77.1	89.81	81.11
7	88.03	86.62	82.3	80.15	**91.43**	91.17
8	94.37	94.37	94.28	86.33	**100**	98.27

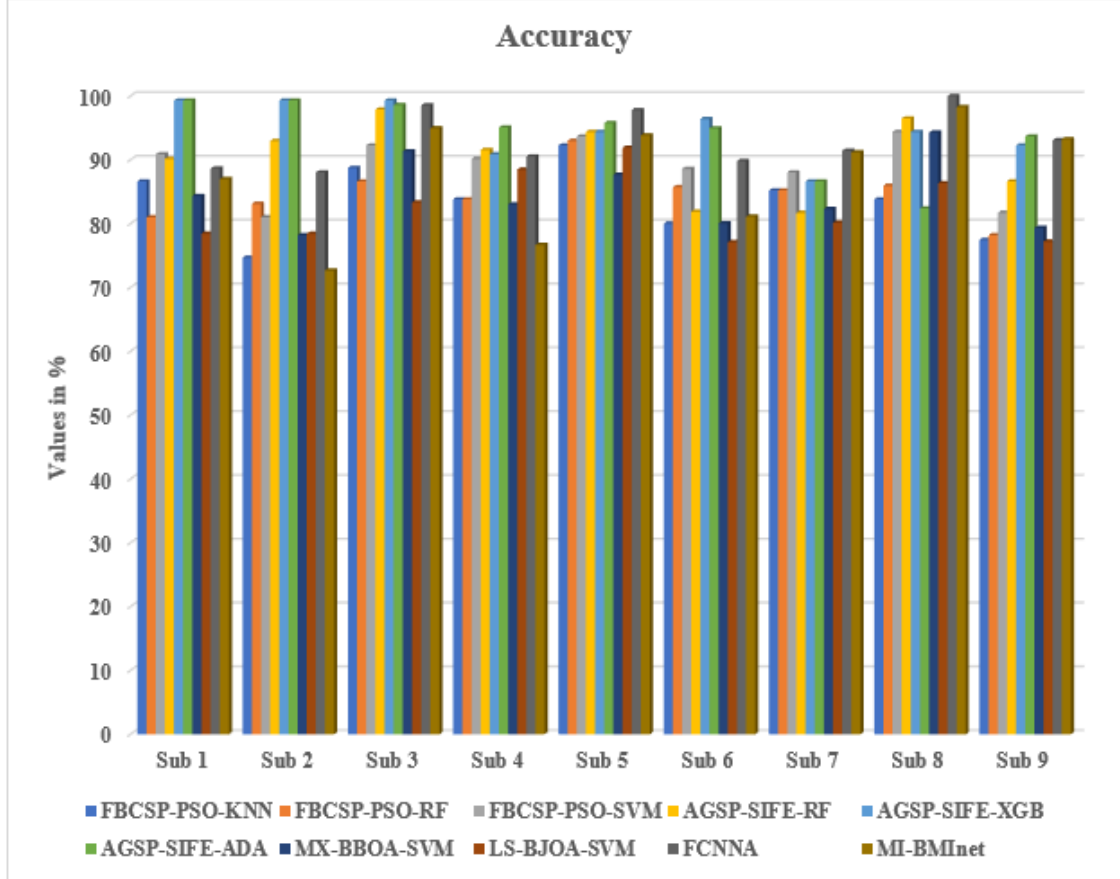| 9 | 81.69 | 92.25 | 79.33 | 77.18 | 93.08 | **93.23** |
| Mean | 88.95 | **94.74889** | 84.49 | 83.59 | 93.09 | 87.65 |

Figure 6.5. Accuracy comparison of various channel selection algorithms

This demonstrates the robustness and generalizability of AGSP-SIFE-XGB across varied EEG signal patterns and conditions. The FBCSP-PSO-SVM method also performs strongly, ranking second in terms of mean accuracy with 88.95%, consistently outperforming existing methods like MX-BBOA-SVM (84.49%) and LS-BJOA-SVM (83.59%). This indicates that integrating Particle Swarm Optimization (PSO) with FBCSP effectively enhances feature selection, contributing to better classification results. Among the existing methods, FCNNA shows competitive performance with a mean accuracy of 93.09%, particularly excelling in subjects 3, 5, and 8 (with 98.54%, 97.78%, and 100% respectively), which suggests its deep learning architecture can effectively capture non-linear EEG signal patterns in certain cases. However, it shows variability across other subjects, lowering its overall reliability.

MI-BMInet, while showing good results in some subjects (especially 3, 5, 8, and 9), has a lower mean accuracy of 87.65%, indicating inconsistency in performance. MX-BBOA-SVM and LS-BJOA-SVM perform moderately well, but neither approach crosses the 90% threshold

consistently, making them less suitable for applications requiring high accuracy. In conclusion, this comparison strongly supports the efficacy and stability of the proposed AGSP-SIFE-XGB method, which not only outperforms all existing algorithms in terms of average accuracy but also delivers highly consistent results among different subjects. This outcome validate that the intelligent feature extraction and selection strategies embedded in our approach offer substantial improvements over both traditional and deep learning-based classifiers.

CHAPTER 7

CONCLUSION AND FUTURE SCOPE

7.1. CONCLUSION:

In this work, we presented an efficient approach for MI categorization using advanced feature extraction and channel selection techniques. Compared to traditional CSP-based feature extraction, our method—based on AGSP—demonstrated superior performance. AGSP performed better than conventional feature extraction approaches by leveraging the brain's network structure, capturing spatial and functional relations between EEG channels. Unlike time or frequency domain techniques that treat signals independently, AGSP treats EEG as signals on graphs—allowing extraction of more informative, structured features that reflect underlying neural dynamics, leading to better classification performance in tasks like motor imagery. For effective channel selection, the SIFE outperformed standard methods like Particle Swarm Optimization (PSO). SIFE utilizes repulsive fuzzy-based initialization to enhance population diversity, while the three-parent crossover and complement mutation ensure a stable trade-off between exploitation and exploration. We evaluated the selected channels using classifiers such as XG-Boost, Ada-Boost, and Random Forest (RF). Among these, XG-Boost achieved the highest performance, while RF lagged slightly behind. Specifically, the AGSP-SIFE framework attained accuracies of 90.39%, 94.74%, and 93.95% using RF, XG-Boost, and Ada-Boost, respectively. Overall, our method outperformed two existing approaches, FCNNA and MI-MBInet, with accuracy improvements of 1.77% and 8.08%, respectively.

7.2. FUTURE SCOPE:

In future research, we aim to extend our study toward multiclass motor imagery classification, which involves distinguishing between multiple types of imagery actions including left hand, right hand, feet and tongue. Unlike binary classification, multiclass scenarios introduce greater complexity due to the overlapping and subtle nature of EEG patterns across different motor tasks. To tackle this challenge, we plan to explore advanced deep learning architectures such as CNNs, RNNs, and hybrid models. Such models are capable of automatically learning hierarchical

and spatiotemporal features from original EEG signal, which can significantly improve classification performance. Additionally, we will consider methods like data augmentation, attention mechanisms, and domain adaptation to improve the architecture's robustness and generalizability across subjects and sessions.

7.3. SOCIAL IMPACT:

The progress of accurate and effective BCI structures for motor imaging has broad social consequences, especially for those with physical disabilities or motor impairments. Such technologies can restore a sense of independence by letting people to control prostheses, wheelchairs, or computers only by thinking. This improves quality of life, encourages inclusion, and decreases dependency on caretakers.

REFERENCES

1. Hwang, H.-J., K. Kwon, and C.-H. Im, Neurofeedback-based motor imagery training for brain–computer interface (BCI). Journal of neuroscience methods, 2009. 179(1): p. 150–156.
2. Ang, K.K. and C. Guan, EEG-based strategies to detect motor imagery for control and rehabilitation. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 2016. 25(4): p. 392–401.
3. Alimardani, M., S. Nishio, and H. Ishiguro, The importance of visual feedback design in BCIs; from embodiment to motor imagery learning. PloS one, 2016. 11(9): p. e0161945.
4. Pichiorri, F., et al., Brain–computer interface boosts motor imagery practice during stroke recovery. Annals of neurology, 2015. 77(5): p. 851–865.
5. Edelman, B.J., B. Baxter, and B. He, EEG source imaging enhances the decoding of complex right-hand motor imagery tasks. IEEE Transactions on Biomedical Engineering, 2015. 63(1): p. 4–14.
6. Berman, B.D., et al., Self-modulation of primary motor cortex activity with motor and motor imagery tasks using real-time fMRI-based neurofeedback. Neuroimage, 2012. 59(2): p. 917–925.
7. Lin, M., et al., A VR-based motor imagery training system with EMG-based real-time feedback for post-stroke rehabilitation. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 2022. 31: p. 1–10.
8. Chaudhary, U., Non-invasive Brain Signal Acquisition Techniques: Exploring EEG, EOG, fNIRS, fMRI, MEG, and fUS, in Expanding Senses using Neurotechnology: Volume 1–Foundation of Brain-Computer Interface Technology. 2025, Springer. p. 25–80.
9. Bak, S. and J. Jeong, User biometric identification methodology via eeg-based motor imagery signals. IEEE Access, 2023. 11: p. 41303–41314.
10. Singh, A.K. and S. Krishnan, Trends in EEG signal feature extraction applications. Frontiers in Artificial Intelligence, 2023. 5: p. 1072801.
11. Musallam, Y.K., et al., Electroencephalography-based motor imagery classification using temporal convolutional network fusion. Biomedical Signal Processing and Control, 2021. 69: p. 102826.
12. Yu, X., et al., A new framework for automatic detection of motor and mental imagery EEG signals for robust BCI systems. IEEE Transactions on Instrumentation and Measurement, 2021. 70: p. 1–12.
13. Altaheri, H., G. Muhammad, and M. Alsulaiman, Physics-informed attention temporal convolutional network for EEG-based motor imagery classification. IEEE transactions on industrial informatics, 2022. 19(2): p. 2249–2258.
14. Mirzaei, S. and P. Ghasemi, EEG motor imagery classification using dynamic connectivity patterns and convolutional autoencoder. Biomedical Signal Processing and Control, 2021. 68: p. 102584.
15. Guerrero-Mendez, C.D., et al., EEG motor imagery classification using deep learning approaches in naïve BCI users. Biomedical Physics & Engineering Express, 2023. 9(4): p. 045029.

16. Huang, E., et al., Classification of motor imagery EEG based on time-domain and frequency-domain dual-stream convolutional neural network. IRBM, 2022. 43(2): p. 107–113.
17. Pattnaik, S., M. Dash, and S. Sabut. DWT-based feature extraction and classification for motor imaginary EEG signals. in 2016 International Conference on Systems in Medicine and Biology (ICSMB). 2016. IEEE.
18. Wang, K., D.-H. Zhai, and Y. Xia. Motor imagination EEG recognition algorithm based on DWT, CSP and extreme learning machine. in 2019 Chinese control conference (CCC). 2019. IEEE.
19. Arpaia, P., et al., Channel selection for optimal EEG measurement in motor imagery-based brain-computer interfaces. International journal of neural systems, 2021. 31(03): p. 2150003.
20. Abdullah, I. Faye, and M.R. Islam, EEG channel selection techniques in motor imagery applications: a review and new perspectives. Bioengineering, 2022. 9(12): p. 726.
21. Baig, M.Z., N. Aslam, and H.P. Shum, Filtering techniques for channel selection in motor imagery EEG applications: a survey. Artificial intelligence review, 2020. 53(2): p. 1207–1232.
22. Li, D., et al., Joint hybrid recursive feature elimination-based channel selection and ResGCN for cross session MI recognition. Scientific Reports, 2024. 14(1): p. 23549.
23. Gulraiz, A., et al., LASSO homotopy-based sparse representation classification for fNIRS-BCI. Sensors, 2022. 22(7): p. 2575.
24. Yang, J., et al., Channel selection and classification of electroencephalogram signals: an artificial neural network and genetic algorithm-based approach. Artificial intelligence in medicine, 2012. 55(2): p. 117–126.
25. Sheoran, P. and J. Saini, Optimizing channel selection using multi-objective FODPSO for BCI applications. Brain-Computer Interfaces, 2022. 9(1): p. 7–22.
26. Alyasseri, Z.A.A., et al., EEG channel selection for person identification using binary grey wolf optimizer. Ieee Access, 2022. 10: p. 10500–10513.
27. Tiwari, A. and A. Chaturvedi, Automatic channel selection using multiobjective X-shaped binary butterfly algorithm for motor imagery classification. Expert Systems with Applications, 2022. 206: p. 117757.
28. Lotte, F. and M. Congedo, EEG feature extraction. Brain–Computer Interfaces 1: Foundations and Methods, 2016: p. 127–143.
29. Naydenov, C., A. Yordanova, and V. Mancheva, Methodology for EEG and reference values of the software analysis. Open Access Macedonian Journal of Medical Sciences, 2022. 10(B): p. 2351–2354.
30. Geethanjali, P., Y.K. Mohan, and J. Sen. Time domain feature extraction and classification of EEG data for brain computer interface. in 2012 9th international conference on fuzzy systems and knowledge discovery. 2012. IEEE.
31. Alam, M.N., M.I. Ibrahimy, and S. Motakabber. Feature extraction of EEG signal by power spectral density for motor imagery based BCI. in 2021 8th International Conference on Computer and Communication Engineering (ICCCE). 2021. IEEE.
32. Peng, P., et al., Seizure prediction in EEG signals using STFT and domain adaptation. Frontiers in Neuroscience, 2022. 15: p. 825434.

33. Meng, Y., et al., A Stepwise Discriminant Analysis and FBCSP Feature Selection Strategy for EEG MI Recognition. International Journal of Advanced Computer Science & Applications, 2024. 15(5).

34. Jaipriya, D. and K. Sriharipriya, Brain computer interface-based signal processing techniques for feature extraction and classification of motor imagery using EEG: A literature review. Biomedical Materials & Devices, 2024. 2(2): p. 601–613.

35. Wei, X., E. Dong, and L. Zhu. Multi-class MI-EEG classification: using FBCSP and ensemble learning based on majority voting. in 2021 China Automation Congress (CAC). 2021. IEEE.

36. Li, Y., et al., A temporal-spectral-based squeeze-and-excitation feature fusion network for motor imagery EEG decoding. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 2021. 29: p. 1534–1545.

37. Mane, R., et al., FBCNet: A multi-view convolutional neural network for brain-computer interface. arXiv preprint arXiv:2104.01233, 2021.

38. Isa, N.E.z.M., et al. The performance analysis of K-nearest neighbors (K-NN) algorithm for motor imagery classification based on EEG signal. in MATEC web of conferences. 2017. EDP Sciences.

39. Ma, Y., et al., Classification of motor imagery EEG signals with support vector machines and particle swarm optimization. Computational and mathematical methods in medicine, 2016. 2016(1): p. 4941235.

40. Zhang, R., et al., A new motor imagery EEG classification method FB-TRCSP+ RF based on CSP and random forest. IEEE Access, 2018. 6: p. 44944–44950.

41. Tiwari, A. and A. Chaturvedi. A multiclass EEG signal classification model using spatial feature extraction and XGBoost algorithm. in 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 2019. IEEE.

42. Hu, J., Automated detection of driver fatigue based on AdaBoost classifier with EEG signals. Frontiers in computational neuroscience, 2017. 11: p. 72.

43. Antony, M.J., et al., Classification of EEG using adaptive SVM classifier with CSP and online recursive independent component analysis. Sensors, 2022. 22(19): p. 7596.

44. An, Y., H.K. Lam, and S.H. Ling, Multi-classification for EEG motor imagery signals using data evaluation-based auto-selected regularized FBCSP and convolutional neural network. Neural Computing and Applications, 2023. 35(16): p. 12001–12027.

45. Kouka, N., et al., EEG channel selection-based binary particle swarm optimization with recurrent convolutional autoencoder for emotion recognition. Biomedical Signal Processing and Control, 2023. 84: p. 104783.

46. Kardam, V.S., S. Taran, and A. Pandey, BSPKTM-SIFE-WST: bispectrum based channel selection using set-based-integer-coded fuzzy granular evolutionary algorithm and wavelet scattering transform for motor imagery EEG classification. Signal, Image and Video Processing, 2025. 19(4): p. 1–16.

47. Dong, X., et al., Graph signal processing for machine learning: A review and new perspectives. IEEE Signal processing magazine, 2020. 37(6): p. 117–127.

48. Sandryhaila, A. and J.M. Moura. Discrete signal processing on graphs: Graph fourier transform. in 2013 IEEE International Conference on Acoustics, Speech and Signal Processing. 2013. IEEE.

49. Tseng, C.-C. and S.-L. Lee. A generalized heat kernel smoothing filter for signal denoising over graph. in 2024 IEEE International Symposium on Circuits and Systems (ISCAS). 2024. IEEE.

50. Kuwil, F.H., A new feature extraction approach of medical image based on data distribution skew. Neuroscience Informatics, 2022. 2(3): p. 100097.

51. Zhang, Y., et al. Exact feature distribution matching for arbitrary style transfer and domain generalization. in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022.

52. Men, X., et al. A new time domain filtering method for calculating the RMS value of vibration signals. in 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA). 2018. IEEE.

53. Gonzalez, J.D.T. and W. Kinsner. Multiscaleanalysis of Skewness for Feature Extraction Inreal-Time. in 2018 IEEE 17th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC). 2018. IEEE.

54. Dwyer, R., Use of the kurtosis statistic in the frequency domain as an aid in detecting random signals. IEEE Journal of Oceanic Engineering, 2003. 9(2): p. 85–92.

55. Saadatmand, H. and M.-R. Akbarzadeh-T, Set-based integer-coded fuzzy granular evolutionary algorithms for high-dimensional feature selection. Applied Soft Computing, 2023. 142: p. 110240.

56. Kumar, B. and D. Gupta, Universum based Lagrangian twin bounded support vector machine to classify EEG signals. Computer methods and programs in biomedicine, 2021. 208: p. 106244.

57. Parmar, S.K., O.A. Ramwala, and C.N. Paunwala. Performance evaluation of svm with non-linear kernels for eeg-based dyslexia detection. in 2021 IEEE 9th region 10 humanitarian technology conference (R10-HTC). 2021. IEEE.

58. Samal, D., et al. EEG Signal Classification for Management of Epilepsy using RBF Kernel-Based Support Vector Machine. in 2024 3rd Odisha International Conference on Electrical Power Engineering, Communication and Computing Technology (ODICON). 2024. IEEE.

59. Su, Y., et al., Implementation of SVM-based low power EEG signal classification chip. IEEE Transactions on Circuits and Systems II: Express Briefs, 2022. 69(10): p. 4048–4052.

60. Zhang, S., et al., Efficient kNN classification with different numbers of nearest neighbors. IEEE transactions on neural networks and learning systems, 2017. 29(5): p. 1774–1785.

61. Ao, Y., et al., The linear random forest algorithm and its advantages in machine learning assisted logging regression modeling. Journal of Petroleum Science and Engineering, 2019. 174: p. 776–789.

62. Gomes, H.M., et al., A survey on ensemble learning for data stream classification. ACM Computing Surveys (CSUR), 2017. 50(2): p. 1–36.

63. Natras, R., B. Soja, and M. Schmidt, Ensemble machine learning of random forest, AdaBoost and XGBoost for vertical total electron content forecasting. Remote Sensing, 2022. 14(15): p. 3547.

64. Chen, Y., R. Chang, and J. Guo, Emotion recognition of EEG signals based on the ensemble learning method: AdaBoost. Mathematical Problems in Engineering, 2021. 2021(1): p. 8896062.
65. Wang, S., et al., A new method of diesel fuel brands identification: SMOTE oversampling combined with XGBoost ensemble learning. Fuel, 2020. 282: p. 118848.

APPENDIX

LIST OF PUBLICATIONS−MANUSCRIPTS

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-42

PLAGIARISM VERIFICATION

Title of the Thesis _____

_____

_____

Total Pages _____

Name of the Scholar_____

Supervisor

(1) _____

Department_____

This is to report that the above thesis was scanned for similarity detection. Process and outcome is given below:

Software used: _____

Similarity Index: _____

Total Word Count: _____

Date: _____

Candidate's Signature   **Signature of Supervisor**

CV

Ashish Karaiya

Final_thesis_EEG_updated_25_05_25 (1).pdf

Delhi Technological University

Document Details

Submission ID

trn:oid:::27535:97677629

Submission Date

May 25, 2025, 11:49 PM GMT+5:30

Download Date

May 25, 2025, 11:52 PM GMT+5:30

File Name

Final_thesis_EEG_updated_25_05_25 (1).pdf

File Size

1.8 MB

59 Pages

16,645 Words

91,509 Characters

11% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

▸ Bibliography

▸ Quoted Text

▸ Cited Text

▸ Small Matches (less than 8 words)

Match Groups

168 Not Cited or Quoted 11%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

7% 🌐 Internet sources

8% 📖 Publications

7% 👤 Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.