Received: December 28, 2024. Revised: March 8, 2025.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Improvement of Motor Imagery Classification in Brain-Computer Interface Based on EEG Signals Analysis

Iman Hussein AL-Qinani[1,2]* **Yarub Alazzawi**[3]

[1]*Informatics Institute for Postgraduate Studies, Iraqi Commission for Computers & Informatics, Baghdad, Iraq*
[2]*Department of Computer Science, College of Education, Mustansiriyah University, Baghdad, Iraq*
[3]*Mechatronics Engineering Department, Al-Khwarizmi College of Engineering, University of Baghdad, Iraq*
* Corresponding author's Email: phd202120685@iips.edu.iq, iman.alqinani@uomustansiriyah.edu.iq

Abstract: Rapid developments in AI applications are a pivotal step for the development of modern technologies, highlighting the importance of brain signal classification as one of the promising areas. This study presents a deep learning-driven one-dimensional convolutional neural networks (1D-CNN) model for improving electroencephalogram (EEG) signals classification. The model is trained on different data balance scenarios to determine their impact on performance, and preprocessing techniques, including normalization and principal component analysis, are applied to reduce dimensions and improve performance. The model achieves high classification accuracy depending on the BNCI Horizon 2020 dataset, with the average performance accuracy for the eight classes reaching 91.01% in training and 90.99% in testing, while for the seven classes, it is 99.6%. The study confirms the effectiveness of using different techniques for preprocessing and deep neural networks in classifying EEG signals, which improves the potential for developing brain-computer interfaces to facilitate communication between humans and technology.

Keywords: Imagery, EEG signals, Classification, Motor, 1D-CNN.

1. Introduction

Brain-computer interfaces (BCI) are an innovative technology that aims to enhance communication between the human brain and computer systems [1]. This technology relies on reading and analyzing brain signals, such as EEG signals, to interpret mental commands and convert them into concrete actions. BCI systems are of great importance in medical applications and assistive technologies, such as helping people with motor disabilities, including controlling prosthetic limbs and wheelchairs or improving the quality of life through brain control interfaces [2-4]. Fig. 1 illustrates the framework of the typical BCI system that consists of multiple stages, including signal acquisition, preprocessing, feature extraction, and classification [5-7].

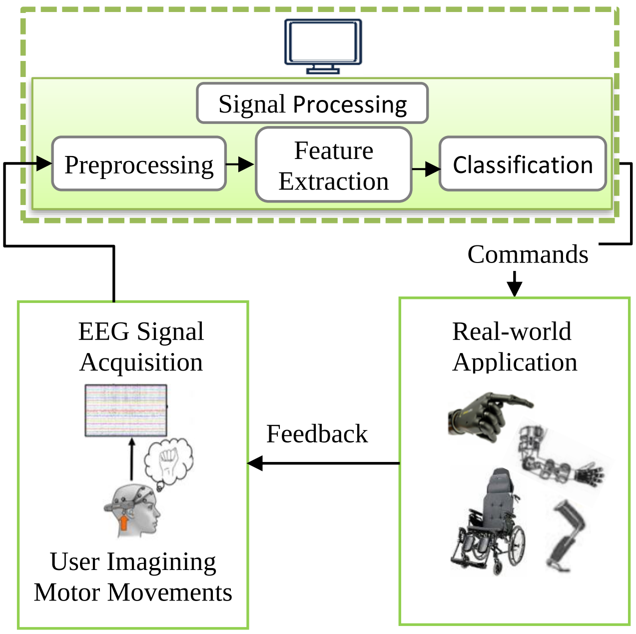


Figure. 1 A Framework of typical BCI system

One of the most critical challenges in BCI systems is accurately distinguishing between imaginary movements (Motor Imagery-MI) that represent complex brain activity. These movements, such as elbow flexion or hand opening, require high data analysis and classification capabilities, especially with the use of multi-channel EEG data. EEG signals are inherently rich in noise and complex due to their reliance on the electrical activity of the brain [8, 9].

In this context, this study relied on the data of the BNCI Horizon 2020 dataset, which is an essential reference in the field of BCI research [10]. This database includes EEG signals representing six movements, the resting state and an unknown class. The data were recorded using 61 EEG channels, which provides a high level of detail but increases the complexity of processing and analysis.

The main contributions of the presented study can be summarized as follows:

- Analyzing EEG signals in an integrated method based on data refinement to reduce complexity and ensure signal quality.
- Develop data balancing scenarios to select the optimal scenarios to improve model performance.
- Incorporating preprocessing scenarios such as normalization and principal component analysis (PCA) to reduce dimensionality and improve model efficiency.
- Using 61 EEG channels to improve classification accuracy and take advantage of the comprehensive information available.
- Multiclass classification using deep learning highlights the potential of deep learning techniques, such as 1D-CNN, to classify MI in a complex environment while adopting multiclass classification to expand the range of commands that the model can handle.
- Improve the model by including an "Unknown class" representing the auxiliary signals accompanying the data recording experience, such as visual and auditory stimuli and time markers, enhancing the model's comprehensiveness in real-world applications.

This paper is organized as follows: Section 2 provides a comprehensive review of the literature related to the topic, Section 3 explains the material and methods used in the work, Section 4 describes the results obtained, Section 5 discusses these results, and finally, Section 6 presents conclusions and future recommendations.

2. Literature review

The study summarizes research related to the BNCI Horizon 2020 database and other databases, focusing on developments in the analysis of EEG signals using artificial intelligence (AI) and deep learning techniques to improve the classification of imaginary movements and expand its applications. Jeong et al. [11] introduced a subject-dependent and section-wise spectral filtering (SSSF) method to improve the decoding performance of movement-related cortical potentials (MRCPs) in brain-machine interfaces (BMI). The method considers individual MRCP characteristics and temporal sections, leading to enhanced accuracy. The study shows successful decoding results in pseudo-online analysis, which is still insufficient for real-world applications due to the noise generated by body movements and exoskeleton vibrations. This could be considered one of the limitations of the study. In addition, the model training time was longer compared to traditional methods, which may limit the efficiency of real-time application. Mammone et al. [12] developed a deep Convolutional Neural Network (DCNN) classifier based on a 3D representation of time-frequency (TF) maps to detect premovement phases by distinguishing between EEG segments preceding motion onset (premovement) and resting states. The approach demonstrated higher performance in classifying premovement vs. rest compared to premovement vs. premovement for different motion types. One of the most critical limitations of the research is the average performance in distinguishing different motor planning movements, which indicates the need for further improvement in processing convergent and complex movements. Bi et al. [13] presented a new model called time-spatial parallel network (TSPNet) for classifying EEG-based MI signals in BCI systems. The model aims to improve the ability to distinguish between different movements by processing temporal and spatial features in parallel. The feature extraction process is divided into three main modules: time dimension feature extractor (TDFE) to extract temporal information from signals, spatial dimension feature extractor (SDFE) to analyze spatial patterns, and time-spatial parallel feature extractor (TSPFE) to remove redundancy between features and improve classification performance. They also developed a feature visualization algorithm based on frequency masking, where the effect of eliminating certain frequency bands on the model's performance was analyzed, which helped identify the most important frequencies for each type of movement. However, the accuracy in classifying movements is still low, which may affect the effectiveness of the model in real-world applications. Batistić et al. [14] proposed using Short-Term Entropy extracted from time-frequency representations (TFRs) to improve the classification of MI using EEG signals. The approach aims to overcome the limitations of conventional features by testing the model on two different datasets, where Shannon and Rényi entropy showed superiority in detecting MI compared to conventional methods. However, the classification of different movement directions (e.g. Rights vs Up) was less accurate than moves classification against the baseline condition (rest), calling for improved feature extraction strategies. Bi et al. [15] presented the transfer data learning network (TDLNet), a deep learning model based on CNNs, to improve the classification of EEG signals for imagining upper motor movements across different participants. The model features a transfer data module (TDM) integrating participants' data to enhance generalization, an inception module that extracts features from the signal by analyzing multi-scale temporal information using convolution kernels of different sizes, and a residual attention mechanism (RAMM) that enhances the most important features in the EEG signals recorded. The model outperformed traditional models such as EEGNet and DeepConvNet in terms of classification accuracy, confirming the effectiveness of the proposed methodology in improving the recognition of imaginary movements. However, the generalization ability of the model remains a challenge due to the large variations in EEG signals across participants. Jia et al. [16] proposed a two-stage training–temporal-spectral neural network (TTSNet), a deep neural network based on temporal-spectral analysis to improve the classification of EEG signals associated with upper limb movements. TTSNet is based on a two-stage training where spectral features are extracted using task-related component analysis (TRCA) analysis and then passed through a CNN to process the temporal information. The method also uses filter banks to extract features from different frequency bands, which helps enhance neural pattern recognition. The results showed that TTSNet outperformed filter bank task-related component analysis (FBTRCA) and EEGNet in binary classification, providing more stable performance and higher accuracy. However, in multi-class classification, the performance was relatively lower, indicating additional challenges in accurately recognizing multiple movements and emphasizing the need for feature extraction and neural signal analysis improvements. The research has some limitations, such as the absence of separate analysis for each subject and the lack of sufficient details about some methodological aspects, which affects the generalization of the results and the accuracy of the evaluation. Considering the challenges faced by previous studies, which often focus on binary classification or a limited number of classes while relying on limited data channels of EEG signals, this study proposes 1D-CNN as an effective solution to overcome these limitations and achieve multi-class classification using multi-channel data with high accuracy and efficiency.

3. Material and methods

This section describes the dataset for EEG signals and the Convolutional Neural Network (CNN) model for classifier MI in BCI.

3.1 Dataset description

This study is based on the "Decoding Upper Limb Movement from EEG (001-2017)" dataset from the BNCI Horizon 2020 project [10, 17], which focuses on EEG recordings of upper limb movements. The dataset was collected from 15 healthy adults aged 22–40, with a gender balance of 6 males and 9 females. A variety of movements were recorded and performed either actually or imagined by the participants, which were recorded via several specialized channels, where EEG signals were recorded via 61 channels covering frontal, central, parietal, and temporal areas to measure the electrical activity of the brain, in addition to electrooculography (EOG) signals via 3 channels to measure eye movements. Sensor gloves across 19 channels and exoskeleton sensors across 13 channels were used to determine movement onset. Each channel has a sequential channel label. Table 1 shows the range of channel numbers used in the experiment and their labels and data types. The experiment included six main types of upper Limb movements: elbow flexion (EF), elbow extension (EE), forearm extension (FS), forearm extension (FP), hand opening (HO), and hand closing (HC). In addition, a rest class was recorded, where participants were instructed to avoid any movement, and auxiliary

Table 1. Channel labels

NO. Channel	Channel Label	Type data
1-61	F3 to PPO2h	EEG electrode
62-64	EOG left to EOG right	EOG positions
65-83	thumb near to -	Data glove sensors
84-96	hand X to -	exoskeleton sensors

Received: December 28, 2024. Revised: March 8, 2025.

Table 2. Event codes

Class	Event Code
EF	0X600
EE	0X601
FS	0X602
FP	0X603
HC	0X604
HO	0X605
rest	0X606

signals occurring during the data recording session were recorded, which we were given the name of the "unknown class". This class represents specific signals in the data, such as a cross on screen and beep, which do not belong to the six basic movements or the rest category and do not carry the event codes shown in Table 2; The signals are coded as events. Table 2 shows the hexadecimal codes assigned to each class of movements used in the experiment, where a specific code is assigned to each type of movement. Participants performed two separate sessions: the first was to execute the movements (known as Motor Execution - ME), and the second was to imagine the movements (known as Motor Imagination - MI). During each session, 10 runs were recorded, each run including 6 trials for each class of recorded movement. Fig. 2 illustrates the dataset description and its components.

It will be split into 60% and 40% for training and testing, respectively, to be used in the proposed model, ensuring an appropriate balance for evaluating the overall performance of the model.

3.2 Convolutional Neural Network (CNN) model

Convolutional Neural Networks (CNN) is an advanced deep learning model that processes temporal and multidimensional data. The network relies on successive layers to analyze the data gradually and extract basic and complex features [18]. It starts with convolutional layers that extract initial patterns from the data, followed by activation layers such as rectified linear unit (ReLU) that add a nonlinearity element to improve the model's ability to learn complex relationships, which is defined as Eq. (1):

$$\sigma(c) = \max(0, c) \tag{1}$$

Where σ represents the activation layers, and c denotes the input. This function yields zero for negative inputs and the input itself for positive values, producing piecewise linear behavior in neural networks.

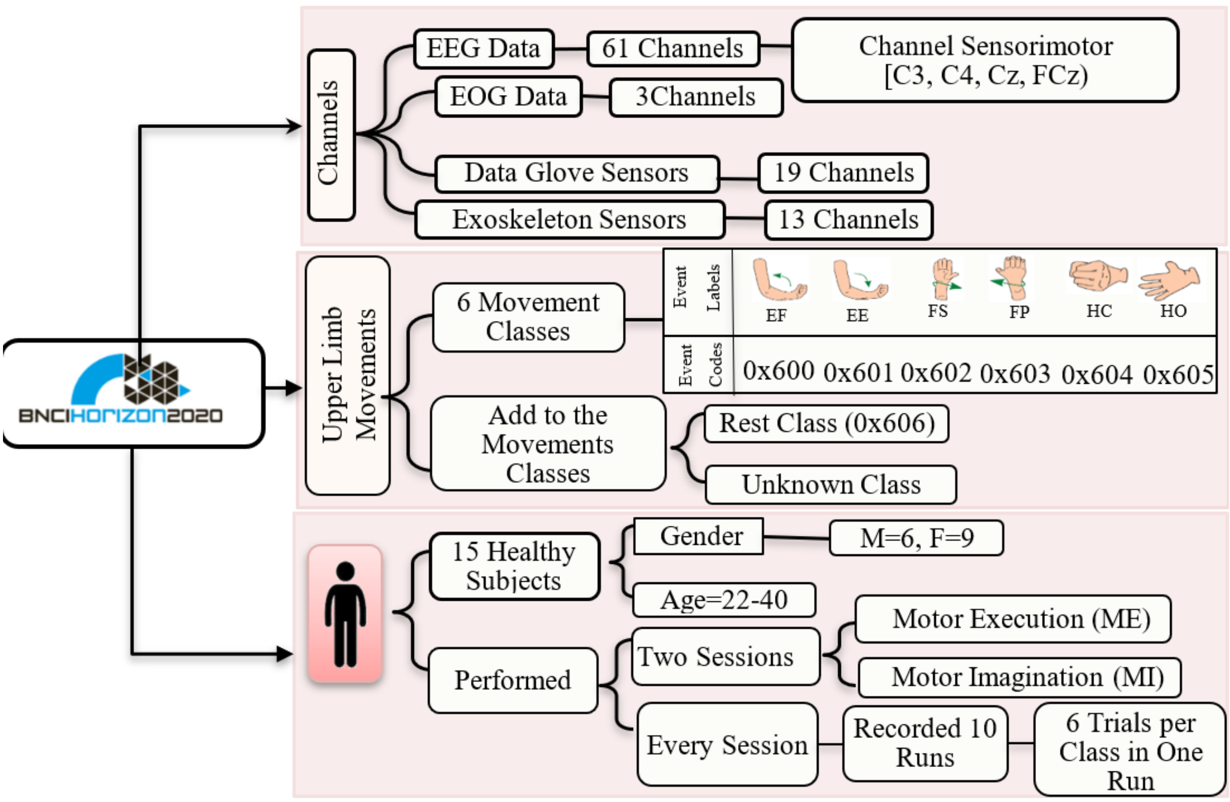


Figure. 2 Dataset description

 DOI: 10.22266/ijies2025.0531.26

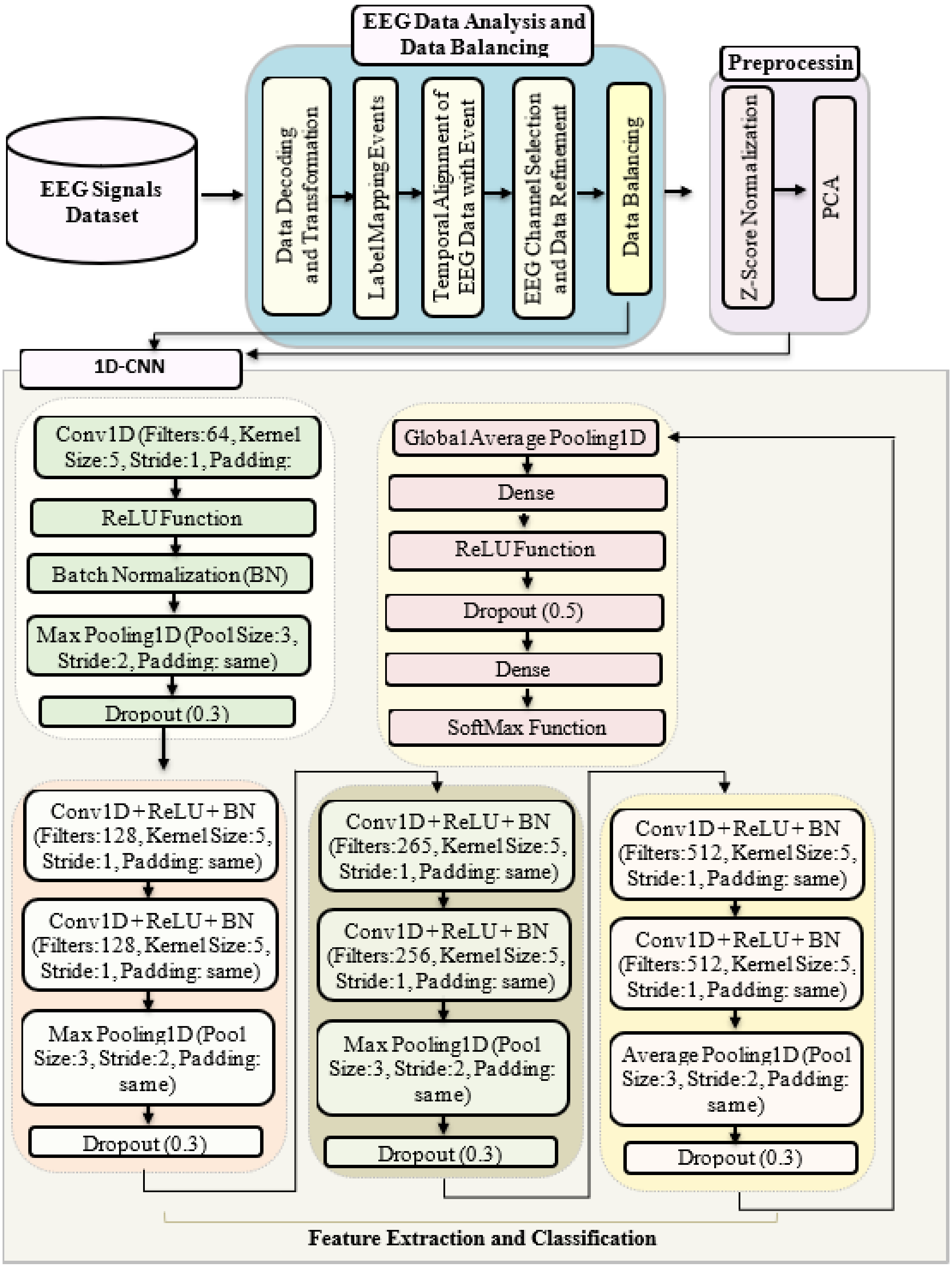


Figure. 3 Proposed model diagram

Then, the Pooling layers reduce the data dimensions, while normalization layers contribute to training stability and speed up the learning process. In addition, dropout layers randomly disable some neural connections during training to avoid overfitting and improve the model's generalization ability. Finally, the network ends with a fully Connected layer in which the extracted features are integrated, and the Softmax function is used to provide multi-class classification, which is defined as Eq. (2) [19-22]:

$$softmax(z)_i = p_i = \frac{\exp{(z_i)}}{\sum_{j=1}^{M} \exp{(z_j)}} \tag{2}$$

Received: December 28, 2024. Revised: March 8, 2025.

Where z_i denotes the i^{th} component of the output vector from the preceding layer z, The numerator is standardized by summating of all exponential terms from 1 to M (M is No. classes) to constrain the value of p_i within the range of 0 and 1.

In this study, 1D-CNN was employed to analyze and classify EEG signals to distinguish MI. The network-assisted extracting of temporal and frequency patterns directly from the raw signals enhances the model's efficiency.

3.3 Proposed model

The model consists of four main stages that aim to analyze and classify brain signals effectively, as shown in Fig. 3. The first stage focuses on analyzing brain signal data extracted from the BNCI 2020 Horizon project. It comprises preparing and processing the data for training and classification processes in subsequent stages. The second stage involves developing several scenarios for data balance. These scenarios were designed to address the challenges associated with the unbalanced distribution of classes within the data, which contributes to improving the model's efficiency. The third stage comprises applying preprocessing scenarios using normalization and PCA techniques. It aims to improve data quality and reduce the computational complexity of the model. Finally, 1D-CNN was customized to effectively classify MI from EEG signals.

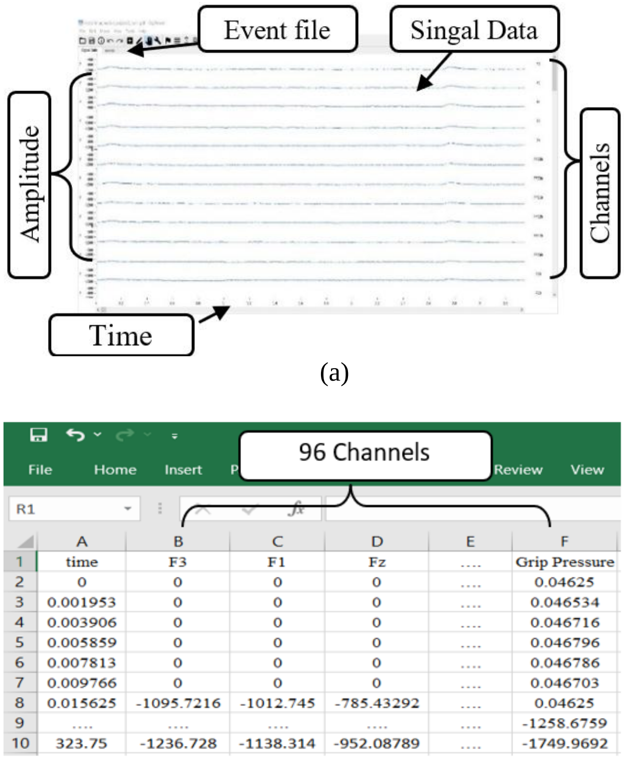
3.3.1. EEG data analysis stage

a) Data decoding and transformation

In this step, the medical EEG files in GDF (General Data Format) format are decoded using the mne library and converted into a readable CSV format, which allows easy access to channels and timing information and flexible data handling, thus making the data organized and ready for use in subsequent analyses, as shown in Fig. 4 , part (a) shows a sample of the data in its original format (GDF) before conversion, while part (b) shows the same sample after conversion (CSV) to facilitate its use in analysis.

b) Extracting and label mapping experimental events information

This step extracts event IDs, event codes and timings from the event file embedded in the GDF file, assigning an event timing as the start of each event (Event starting/Sample Starting) and converting event codes from hexadecimal to decimal to facilitate assigning labels to each movement based on numeric values of the events. A map is created that associates



	A	B	C	D	E	F
1	time	F3	F1	Fz	....	Grip Pressure
2	0	0	0	0	....	0.04625
3	0.001953	0	0	0	....	0.046534
4	0.003906	0	0	0	....	0.046716
5	0.005859	0	0	0	....	0.046796
6	0.007813	0	0	0	....	0.046786
7	0.009766	0	0	0	....	0.046703
8	0.015625	-1095.7216	-1012.745	-785.43292	....	0.04625
9	....	....	....	....	....	-1258.6759
10	323.75	-1236.728	-1138.314	-952.08789	....	-1749.9692

(b)

Figure. 4 Sample of data decoding and transformation: (a) GDF file and (b) CSV file

each event code with a specific movement, such as EF or HO. Codes that do not match known movements and are not listed in Table 2 are classified as "unknown" class. Fig. 5 shows a sample of the process of linking events to their labels. Fig. 5(a) shows a sample of the event file before processing, while Fig. 5(b) highlights the results after assigning events to their labels according to the main movements.

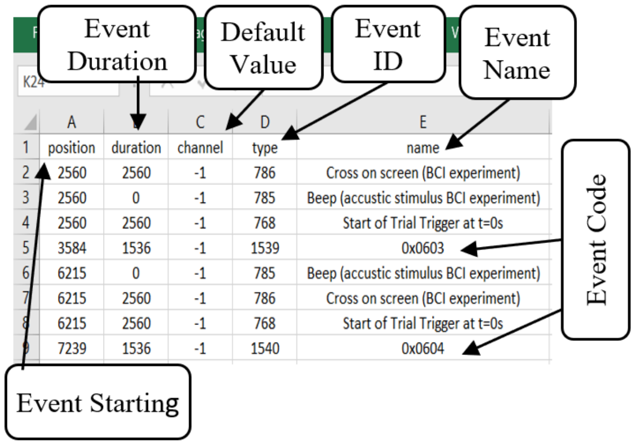
c) Temporal alignment of EEG data with event annotations

In this step, the brain signal recording data from the EEG data file (file product from section 3.3.1(a)) are merged with the event data from the event file (file product from section 3.3.1(b)) in temporally aligning to ensure accurate synchronization between signals and events. The event data (sample starting) in the event file is converted to the time units in seconds using the following Eq. (3):

$$\text{Time} = \frac{\text{Start Sample}}{\text{Sampling Rate}} \quad (3)$$

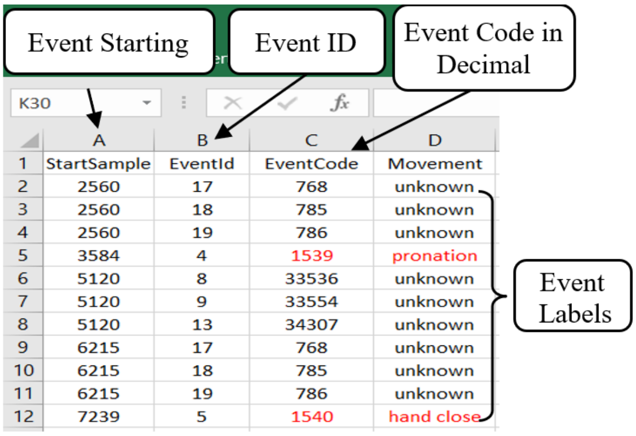
Where the sampling rate used is 512 samples per second. The signal data is then merged with the event data based on the closest time point to each signal,

Received: December 28, 2024. Revised: March 8, 2025.



	A	B	C	D	E
1	position	duration	channel	type	name
2	2560	2560	-1	786	Cross on screen (BCI experiment)
3	2560	0	-1	785	Beep (accustic stimulus BCI experiment)
4	2560	2560	-1	768	Start of Trial Trigger at t=0s
5	3584	1536	-1	1539	0x0603
6	6215	0	-1	785	Beep (accustic stimulus BCI experiment)
7	6215	2560	-1	786	Cross on screen (BCI experiment)
8	6215	2560	-1	768	Start of Trial Trigger at t=0s
9	7239	1536	-1	1540	0x0604

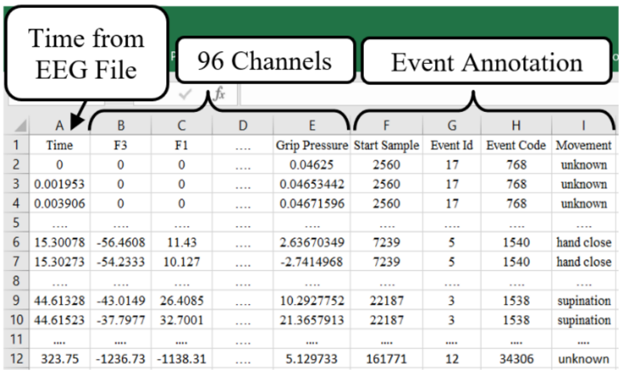
(a)



	A	B	C	D
1	StartSample	EventId	EventCode	Movement
2	2560	17	768	unknown
3	2560	18	785	unknown
4	2560	19	786	unknown
5	3584	4	1539	pronation
6	5120	8	33536	unknown
7	5120	9	33554	unknown
8	5120	13	34307	unknown
9	6215	17	768	unknown
10	6215	18	785	unknown
11	6215	19	786	unknown
12	7239	5	1540	hand close

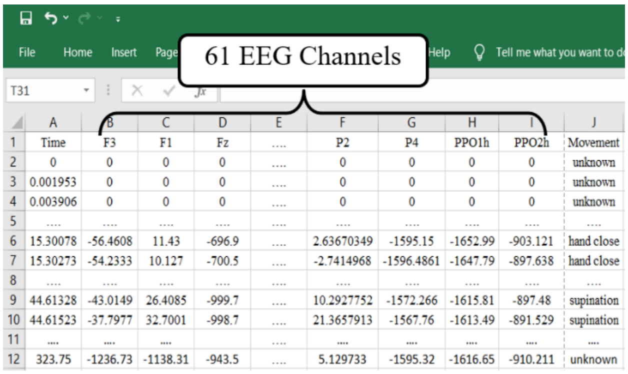
(b)

Figure. 5 Sample of event processing and mapping: (a) sample of event file and (b) sample of event-label mapping



	A	B	C	D	E	F	G	H	I
1	Time	F3	F1	....	Grip Pressure	Start Sample	Event Id	Event Code	Movement
2	0	0	0	....	0.04625	2560	17	768	unknown
3	0.001953	0	0	....	0.04653442	2560	17	768	unknown
4	0.003906	0	0	....	0.04671596	2560	17	768	unknown
5	....	....	....	....	....	....	....	....	....
6	15.30078	-56.4608	11.43	....	2.63670349	7239	5	1540	hand close
7	15.30273	-54.2333	10.127	....	-2.7414968	7239	5	1540	hand close
8	....	....	....	....	....	....	....	....	....
9	44.61328	-43.0149	26.4085	....	10.2927752	22187	3	1538	supination
10	44.61523	-37.7977	32.7001	....	21.3657913	22187	3	1538	supination
11	....	....	....	....	....	....	....	....	....
12	323.75	-1236.73	-1138.31	....	5.129733	161771	12	34306	unknown

Figure. 6 Sample of aligned EEG data with event annotations



	A	B	C	D	E	F	G	H	I	J
1	Time	F3	F1	Fz	....	P2	P4	PPO1h	PPO2h	Movement
2	0	0	0	0	....	0	0	0	0	unknown
3	0.001953	0	0	0	....	0	0	0	0	unknown
4	0.003906	0	0	0	....	0	0	0	0	unknown
5	....	....	....	....	....	....	....	....	....	....
6	15.30078	-56.4608	11.43	-696.9	....	2.63670349	-1595.15	-1652.99	-903.121	hand close
7	15.30273	-54.2333	10.127	-700.5	....	-2.7414968	-1596.4861	-1647.79	-897.638	hand close
8	....	....	....	....	....	....	....	....	....	....
9	44.61328	-43.0149	26.4085	-999.7	....	10.2927752	-1572.266	-1615.81	-897.48	supination
10	44.61523	-37.7977	32.7001	-998.7	....	21.3657913	-1567.76	-1613.49	-891.529	supination
11	....	....	....		....	....	....	....	....	....
12	323.75	-1236.73	-1138.31	-943.5	....	5.129733	-1595.32	-1616.65	-910.211	unknown

Figure. 7 Sample of EEG channel selection and data refinement

ensuring that each data point is associated with the corresponding class. The resulting file contains information, including the timing of the recorded signals, 96 channels of signal data, as well as event information (Event Annotations) such as Start Sample, Event ID, Event Code, and Movement, which illustrates how the temporal data between the two files is aligned. Fig. 6 shows a sample of the resulting data, where the time values extracted from the EEG file, the recorded channels and the associated event information are displayed, reflecting the synchronization between the neural signals and the recorded activities.

d) EEG channel selection and data refinement

This step focuses on processing the EEG data by selecting the relevant channels, which are 61 EEG channels while excluding unimportant channels such as EOG channels, glove sensors, and exoskeleton sensors. The data is refined by removing rows that only contain the "Time" column with the label "unknown" and no channel-related data, as well as by excluding rows that contain zero values in the selected columns. As shown in Fig. 7, a sample of the resulting file is illustrated after the channel selection and data refining process.

3.3.2. Developing data balance scenarios stage

Designing data balance scenarios is a critical step in handling the imbalance in the distribution of classes within the dataset. Given the dominance of the unknown class, which significantly outnumbers other classes, four different scenarios were developed to mitigate the impact of this disparity and ensure a fair distribution of data for training the model. Table 3 illustrates the details of data balance scenarios across 15 Subjects.

a) No Balance:

In this scenario, all available data is used without any modifications to the distribution. The model is trained on the natural data distribution, where the unknown class dominates, containing approximately 14 times more samples than other classes.

Table 3. Data balance scenarios across 15 subjects

Classes	Data Balance Scenarios			
	No Balance (15 Subjects)	**Full Balance (15 Subjects)**	**3X Unknown (15 Subjects)**	**50% Unknown (15 Subjects)**
Unknown	16884150	1152000	3456000	8064000
Rest	1152000	1152000	1152000	1152000
FS	1152000	1152000	1152000	1152000
FP	1152000	1152000	1152000	1152000
EF	1152000	1152000	1152000	1152000
EE	1152000	1152000	1152000	1152000
HC	1152000	1152000	1152000	1152000
HO	1152000	1152000	1152000	1152000
Total number of samples	24948150	9216000	11520000	16128000

b) Full Balance:

This scenario reduces the number of samples in all classes to match the minimum count available among the other classes. In this case, all classes are equalized to contain the same number of samples, ensuring a fair training process where no class dominates.

c) 3X Unknown Balance:

This scenario adjusts the number of unknown samples to three times the minimum count of the other classes. This approach achieves a moderate balance, allowing the inclusion of more samples from the unknown class while maintaining a reasonable representation for all classes.

d) 50% Unknown Balance:

In this scenario, the unknown samples are reduced to approximately half of their original count. This adjustment reduces the overwhelming dominance of the unknown class and ensures a balanced representation of the other classes.

3.3.3. Preprocessing stage

In this stage, different techniques are used to improve classification performance, such as normalization and dimensionality reduction via PCA. Classification of EEG signals is a significant challenge in the development of prosthetic control systems, as they are high-dimensional and non-stationary in nature, requiring procedures for data preparation and reducing their complexity [23].

a) Z-Score Normalization:

it is used to standardize the range of values in the data, making it more suitable for machine learning models. Z-Score helps identify the outliers by distributing the data in a bell-curve fashion. The normalization is applied as Eq. (4) [23-25]:

$$Zscore = \frac{X - \mu}{\sigma} \tag{4}$$

Where Z is the normalized data, X is the original data, μ is the mean, and σ is the standard deviation computed over X.

b) Principal Component Analysis (PCA):

it is a dimensionality reduction method in which data is transformed into a new space of uncorrelated vectors (principal components) while preserving as much variance in the data as possible [26-30].

In this work, three scenarios were used to prepare the data before training the model:

- Raw Data: Train the model using the original data without making any changes.
- Normalization: Normalize the data range using the Z-Score Normalization method.
- Normalization + PCA: In this scenario, normalization was applied first, then dimensionality was reduced using PCA.

3.3.4. Feature extraction and classification stage

This study customized 1D-CNN to classify EEG signals related to MI. The model is based on a seven-layer architecture designed to analyze raw signals and extract temporal and frequency patterns with high accuracy. The network starts with convolutional layers designed to extract raw and complex features from signals. Seven convolutional layers were used with ReLU activation after each layer to improve the network's ability to learn from nonlinear patterns. Also, five pooling layers are applied to reduce the data size while preserving the most important features, which improves computational efficiency and reduces the complexity of the model. Five Dropout layers were included in the architecture to ensure performance stability and reduce the risk of overfitting. At the penultimate stage, a Global Average Pooling layer reduces each extracted feature to its global mean, reducing the data to a simple and efficient form for handling the final classification. In the final stage, two Dense layers were included: the first reinforces important patterns of the extracted features to facilitate model learning. In contrast, the final layer uses the SoftMax function to provide the final predictions for different classes.

3.3.5. Classification types and training scenarios

This section focuses on the classification scenarios, training methods, and analysis approaches applied to EEG data. Additionally, it includes strategies for selecting channels, which play a critical role in analyzing movement data. A concise breakdown of each aspect is provided:

a) Training Scenarios

- Subject-by-Subject: In this approach, the model is trained and tested using data from a single subject. This method emphasizes individual patterns and allows for personalized analysis.
- All Subjects Together: In this method, the model is trained and tested using combined data from all participants. This scenario creates a generalized model to detect patterns shared across individuals.

b) Classification Types

- Binary Classification: This type focuses on distinguishing between two classes and includes the following Scenarios:
  - mov vs. mov: Classifying one movement against another.
  - mov vs. rest: Classifying a single movement against the rest state.
  - all moves vs. rest: Classifying all movement classes combined against the rest.
- Multiclass Classification: This type distinguishes between multiple classes and includes scenarios such as:
  - 8-class: Includes six movements, rest, and unknown.
  - 7-class: Includes six movements and rest, excluding unknown.
  - 6-class: Includes six movements only, excluding rest and unknown.

c) Data Analysis Approaches

- Single Time Points: EEG data is analyzed using individual samples at specific time points.
- Time Windows: EEG signals are divided into time windows to analyze dynamic changes in brain activity over a specific period.

d) EEG Channels Selection Approaches

Different approaches are employed for selecting EEG channels when analyzing recorded movement data:

- Single Channel (e.g., Cz): This method simplifies the analysis by focusing on activity from motor or sensorimotor regions, reducing complexity.
- Muli-Channels: Utilizes Multiple EEG channels to enhance spatial resolution and provide broader coverage. This Strategy includes:
  - Four Channels (e.g., FCz, C3, Cz, C4): focuses on the sensorimotor region to capture movement-related signals with better spatial resolution.
  - 61 Channels: Ensure comprehensive coverage of brain regions, offering a detailed analysis of movements.

e) Unknown Class

The unknown class represents a set of auxiliary signals that appear during the data recording session, such as:

- Cross on Screen: a visual cue appears on the screen during the BCI experiment.
- Beep (Acoustic Stimulus): An audio cue during the experiment.
- Start of Trial Trigger: marks the start of a trial at t=0.

These signals are accompanying actions for the experiment, providing visual and auditory stimuli or time markers used during recording sessions. Since it does not belong to the six basic movements or the resting class, it is classified as a separate class to ensure the model's comprehensiveness and ability to handle all signals recorded during the experiment.

The addition of this class reflects a thoughtful design aimed at improving the application of the model in real-world environments, such as prosthetic limb control, where processing all recorded signals is essential to ensure efficient and accurate performance of the model under different operating conditions.

4. Results

In this study, we will present a 1D-CNN evaluation model for all scenarios with data balancing and preprocessing.

4.1 Evaluation model

There are several evaluation metrics to evaluate the performance of a model. These metrics include Accuracy, Precision, Recall, and F1-Score [31-34], as shown below:

Accuracy (ACC) is used to evaluate the model's overall performance, as shown in Eq. (5):

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{Total Samples(TP+TN+FP+FN)}} \quad (5)$$

True Positives (TP), True Negatives (TN), False Positives (FP), False Negatives (FN)

Precision (Pr.) measures how well the model predicts only positive items, as illustrated in Eq. (6):

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (6)$$

Recall (Re.) measures the model's ability to detect all true positive cases; it is defined in Eq. (7):

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad (7)$$

F1-Score provides a balance between Precision and Recall, as clarified in Eq. (8):

$$\text{F1} - \text{Score} = \frac{\text{Precision} \times \text{Recall}}{\text{Precision+ Recall}} \times 2 \quad (8)$$

4.2 Hyperparameter tuning

Tuning the model hyperparameter is essential to ensure the optimal performance of the 1D-CNN to classify imaginary movements. The hyperparameters in this work were tuned based on the 3X Unknown data balance scenario to improve the accuracy and stability of the model. Table 4 illustrates the effect of different parameters that were applied, including the learning rate, optimization algorithms, and batch size.

4.3 Data balance scenarios evaluation

In this study, we will evaluate the effect of the original data distribution, which may introduce a bias in the model's performance toward the dominant class, as shown in Fig.8, which shows the distribution of samples across different data balance scenarios using data from 15 participants. The significant difference in the number of samples in the “unknown” class compared to the other classes when no balance is applied is evident, with its samples significantly outnumbering the samples in the remaining classes. Fig.8 reflects the effect of each of the following scenarios (Full balance, 3X Unknown, and 50% Unknown) in reducing the dominance of the “unknown” class and achieving a more even distribution across the different classes of movements.

4.4 Model classification performance evaluation

In this study, we will evaluate the classification performance of the model by applying data balancing, normalization, and PCA scenarios.

Table 4. Hyperparameter tuning to train the 1D-CNN on 3X Unknown data balance scenario

Cases	Optimizer	Learning rate	Batch size	Epochs	Early Stopping	Epoch/sec	Train Accuracy	Test Accuracy
Case 1	Adam	0.001	224	50	no	13	0.84	0.84
Case 2	Adam	0.01	128	50	21	19	0.29	0.3
Case 3	AdamW	0.001	224	50	no	13	0.84	0.84
Case 4	AdamW	0.01	128	50	no	20	0.47	0.47

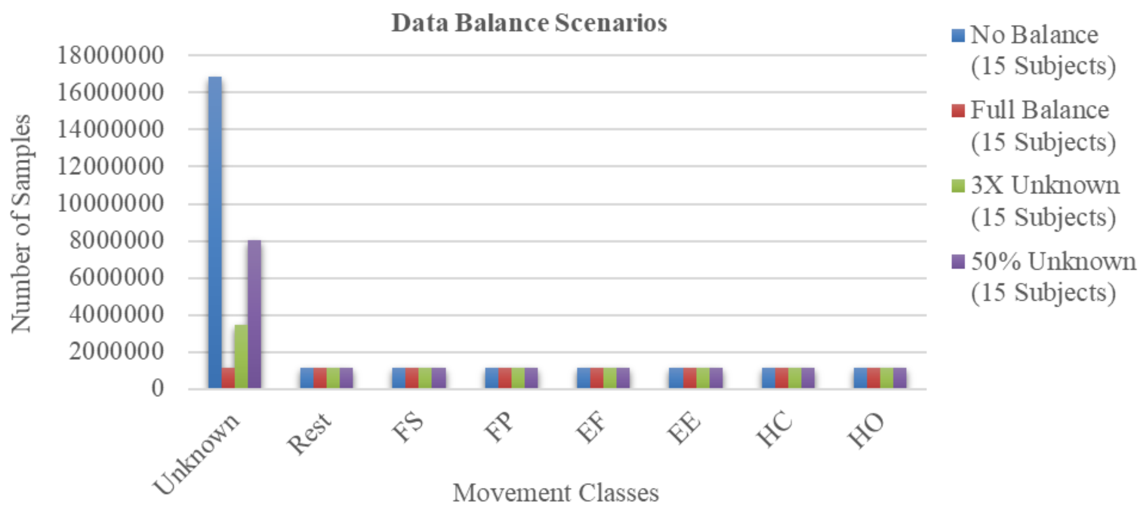


Figure. 8 Data balance scenarios for classes across 15 subjects

4.4.1. Model classification performance evaluation of data balance scenarios

The model's performance was tested using five data balancing scenarios (No Balance, Full Balance, 3X Unknown, 50% Unknown, and Without Unknown), as shown in Tables 5 and 6. This study included a multiclass classification approach using a single time point and analyzing 61 channels of EEG signals. Data balancing scenarios tests were performed on a single subject's data (S_1).

The confusion matrix, evaluation accuracy, and loss show the 3X Unknown scenario as the best data balancing in Figs. 9 and 10, respectively.

Table 5 shows the accuracy evaluation of the proposed model (1D-CNN) in classifying MI across different data balance scenarios. The table shows the number of classes used, the number of training epochs, and the application of the early stopping technique, in addition to the accuracy of the model in the training and testing phases for each scenario.

Table 6 shows the performance evaluation results of the 1D-CNN model when tested on four different

Table 5. Accuracy evaluation of the proposed model (1D-CNN) in classifying MI of data balance scenarios

Data Balance Scenarios	Classes Count	Epochs	Early Stopping	Accuracy Training	Accuracy Testing
No Balance	8	50	21	0.6774	0.6765
		100	21	0.6774	0.6765
		150	N/A	N/A	N/A
Full Balance	8	50	full	**0.8536**	**0.8539**
		100	full	0.8417	0.8425
		150	105	8539	0.8535
3X Unknown	8	50	full	0.8488	0.8467
		100	71	0.8362	0.8333
		150	78	0.8466	0.8441
50% Unknown	8	50	full	0.5065	0.5074
		100	full	0.5126	0.5134
		150	48	**0.5021**	**0.5029**
Without Unknown	7	50	full	**0.9923**	**0.9924**
		100	N/A	N/A	N/A
		150	N/A	N/A	N/A

Table 6. Results of precision, recall, and F1-score for evaluation of the model in classifying MI of data balance scenarios

Classes	No Balance (8 class)			Full Balance (8 class)			3X Unknown (8 class)			50% Unknown (8 class)			Without Unknown (7 class)		
	Pr.	Re.	F1.	Pr.	Re.	F1.	Pr.	Re.	F1.	Pr.	Re.	F1.	Pr.	Re.	F1.
EE	0.00	0.00	0.00	0.85	0.96	0.90	0.84	0.96	0.90	0.79	0.13	0.23	0.99	1.00	1.00
EF	0.00	0.00	0.00	0.84	0.99	0.91	0.84	0.95	0.89	0.00	0.00	0.00	1.00	0.99	0.99
HC	0.00	0.00	0.00	0.84	0.98	0.91	0.81	0.97	0.88	0.00	0.00	0.00	0.99	0.99	0.99
HO	0.00	0.00	0.00	0.84	0.98	0.90	0.85	0.97	0.91	0.00	0.00	0.00	1.00	0.99	0.99
FP	0.00	0.00	0.00	0.84	0.98	0.91	0.83	0.97	0.89	0.00	0.00	0.00	0.99	1.00	0.99
FS	0.00	0.00	0.00	0.88	0.96	0.92	0.85	0.94	0.90	0.00	0.00	0.00	0.99	0.98	0.99
rest	0.00	0.00	0.00	0.88	0.97	0.92	0.84	0.97	0.90	0.00	0.00	0.00	0.99	1.00	0.99
unknown	0.68	1.00	0.81	0.67	0.01	0.03	0.88	0.58	0.70	0.50	1.00	0.67	N/A	N/A	N/A
Macro avg.	0.08	0.12	0.10	0.83	0.85	0.80	0.84	0.91	0.87	0.16	0.14	0.11	0.99	0.99	0.99
Weight avg.	0.46	0.68	0.55	0.83	0.85	0.80	**0.85**	**0.85**	**0.84**	0.31	0.51	0.35	**0.99**	**0.99**	**0.99**

Received: December 28, 2024. Revised: March 8, 2025.

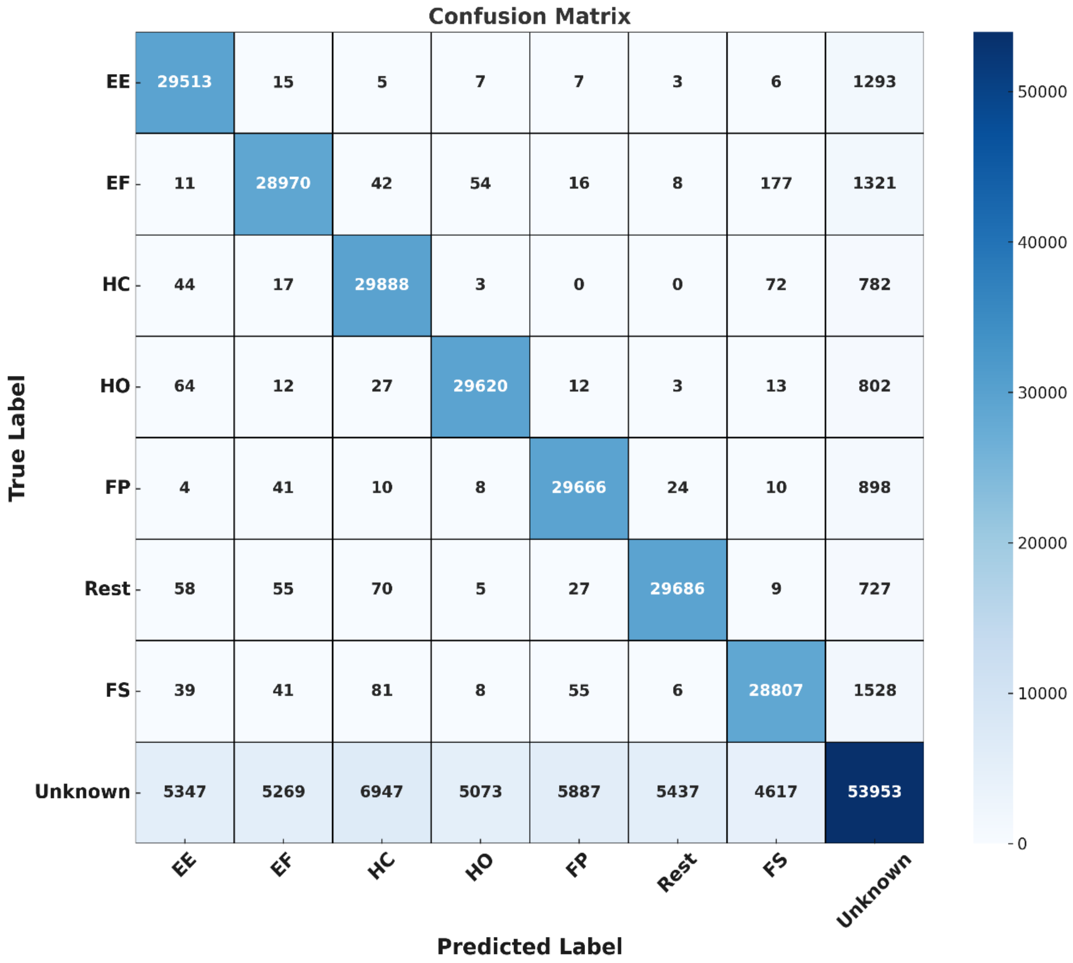


Figure. 9 Confusion matrix for testing the 1D-CNN model on the 3X Unknown data balance scenario using 50 epochs

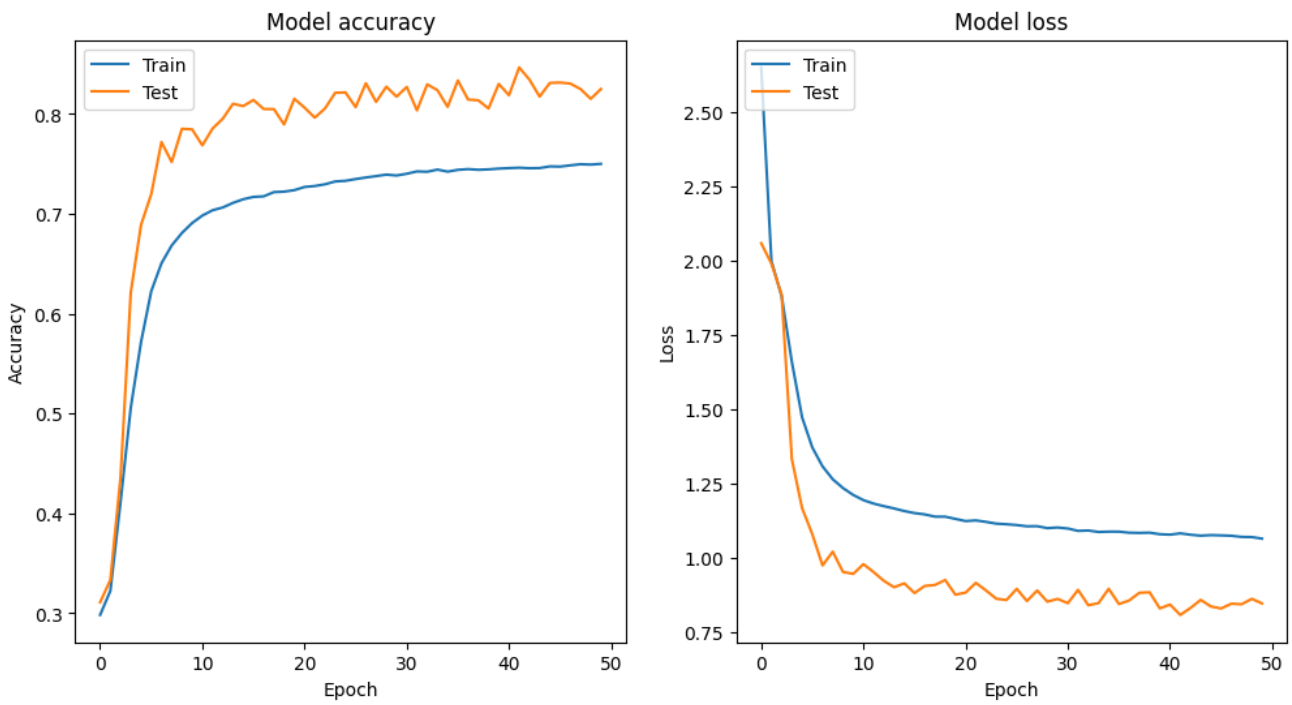


Figure. 10 Performance curves of accuracy and loss for training and testing the 1D-CNN model on the 3X Unknown data balance scenario using 50 epochs

 DOI: 10.22266/ijies2025.0531.26

data balance scenarios using 50 epochs. The table shows the precision, Recall, and F1-Score metrics for each of the classified classes, as well as the overall arithmetic mean (Macro avg.) and the weighted mean (Weight avg.), which helps in comparing the impact of each scenario on the model's performance.

Fig.9 shows the confusion matrix representing the results of testing the proposed model (1D-CNN) on the 3X Unknown scenario using 50 epochs, which is the scenario that showed the best performance among all data balance scenarios. The matrix shows the actual and expected distribution of classes, reflecting the model's ability to distinguish between different classes with high accuracy. While achieving a proper balance between classes and efficiently classifying the unknown class.

Fig.10 shows the curves that represent the performance of the proposed model (1D-CNN) in terms of accuracy and loss during training and testing on the 3X Unknown scenario using 50 epochs. The accuracy curve shows a gradual increase in performance, reflecting the model's ability to adapt to the data, while the loss curve shows a steady decrease, indicating the stability of training and improvement in overall performance.

4.4.2. Model classification performance evaluation of preprocessing scenarios

After selecting the 3X Unknown scenario as the best scenario for data balance due to its high accuracy and having a larger number of samples from the unknown class, three preprocessing techniques were applied to the 3X Unknown data scenario (Raw Data, Normalization, and Normalization with PCA), as described in Section 3.3.3. The model was trained and tested on Subject-by-Subject (S_1 to S_{15}), as shown in Tables 7, 8, 9, and 10.

Table 7 shows the classification accuracy of the 1D-CNN model when applied to the 3X Unknown scenario using 50 epochs across 15 individuals for 8 classes. The table includes the effect of three preprocessing scenarios, namely raw data, normalization, and Normalization + PCA (normalization with dimensionality reduction), on the performance of the model in the training and testing phases.

Table 8 shows the weighted average of the Precision, Recall, and F-Score performance metrics of the proposed model (1D-CNN) when testing the 3X Unknown scenario on different preprocessing techniques on the raw data, normalization, and normalization with (PCA) to classify 8 classes across 15 subjects using 50 epochs.

Table 7. Classification accuracy evaluation of the proposed model (1D-CNN) using 3x Unknown scenario with different preprocessing scenarios applied (Raw data, Normalization, Normalization with PCA) to classify 8 classes of MI across 15 subjects using 50 training epochs

Subjects (8 class)	**Raw Data**		**Normalization**		**Normalization + PCA**	
	Training Phase	Testing Phase	Training Phase	Testing Phase	Training Phase	Testing Phase
S_1	0.8487	0.8467	0.9130	0.9107	**0.9172**	**0.9161**
S_2	0.6733	0.6703	0.9127	0.9116	0.9006	0.8991
S_3	0.3699	0.3703	0.9218	0.9206	0.9205	0.9194
S_4	0.3912	0.3933	0.8429	0.8406	**0.8855**	**0. 8825**
S_5	0.3002	0.3023	0.9210	0.9192	**0.9393**	**0.9375**
S_6	0.3002	0.3023	0.8736	0.8714	**0.8896**	**0.8882**
S_7	0.3388	0.3407	0.8764	0.8744	**0.8983**	**0.8958**
S_8	0.3002	0.3021	0.9134	0.9123	0.9030	0.9025
S_9	0.3072	0.3093	0.9044	0.9031	**0.9190**	**0.9169**
S_{10}	0.6294	0.6296	0.8984	0.8956	0.8861	0.8833
S_{11}	0.3405	0.3420	0.9093	0.9088	**0.9181**	**0.9160**
S_{12}	0.3714	0.3733	0.9087	0.9076	**0.9225**	**0.9208**
S_{13}	0.7156	0.7130	0.9231	0.9227	**0.9294**	**0.9274**
S_{14}	0.3004	0.3024	0.9362	0.9359	0.9190	0.9181
S_{15}	0.3017	0.3033	0.8872	0.8849	**0.9041**	**0.9019**
avg.	**0.4326**	**0.4334**	0.9028	0.9013	**0.9101**	**0.9099**

Table 8. Weighted average of the precision, recall, and F1-Score performance metrics of the proposed model using the 3X Unknown scenario with preprocessing techniques to classify 8 classes

Subjects	**Data Preprocessing Scenarios (8 classes)**								
	Raw Data			**Normalization**			**Normalization + PCA**		
	Pr.	**Re.**	**F1.**	**Pr.**	**Re.**	**F1.**	**Pr.**	**Re.**	**F1.**
S_1	0.85	0.85	0.84	0.91	0.91	0.91	0.92	0.92	0.91
S_2	0.68	0.67	0.67	0.91	0.91	0.91	0.90	0.90	0.90
S_3	0.54	0.37	0.32	0.92	0.92	0.92	0.92	0.92	0.92
S_4	0.54	0.39	0.37	0.85	0.84	0.83	0.88	0.88	0.88
S_5	0.19	0.30	0.14	0.92	0.92	0.92	0.94	0.94	0.94
S_6	0.17	0.30	0.14	0.88	0.87	0.86	0.89	0.89	0.88
S_7	0.59	0.34	0.24	0.88	0.87	0.87	0.90	0.90	0.89
S_8	0.13	0.30	0.15	0.91	0.91	0.91	0.91	0.90	0.90
S_9	0.19	0.31	0.17	0.91	0.90	0.90	0.92	0.92	0.91
S_{10}	0.63	0.63	0.61	0.90	0.90	0.89	0.89	0.88	0.88
S_{11}	0.43	0.34	0.25	0.91	0.91	0.91	0.92	0.92	0.91
S_{12}	0.57	0.37	0.31	0.91	0.91	0.90	0.92	0.92	0.92
S_{13}	0.71	0.71	0.70	0.93	0.92	0.92	0.93	0.93	0.93
S_{14}	0.16	0.30	0.14	0.94	0.94	0.93	0.92	0.92	0.92
S_{15}	0.15	0.30	0.15	0.89	0.88	0.88	0.90	0.90	0.90
avg.	**0.4353**	**0.432**	**0.3467**	0.9047	0.9007	0.8973	**0.9107**	**0.9093**	**0.906**

Table 9. Classification accuracy of the D-CNN model when testing the 3X Unknown scenario on different preprocessing techniques for classifying 7 classes of MI across 15 subjects using 50 epochs

Subjects (7 class)	**Raw Data**		**Normalization**		**Normalization + PCA**	
	Training Phase	Testing Phase	Training Phase	Testing Phase	Training Phase	Testing Phase
S_1	0.9943	0.9937	0.9970	0.9968	0.9949	0.9947
S_2	0.1477	0.1475	0.9982	0.9981	0.9969	0.9968
S_3	0.8883	0.8876	0.9984	0.9984	0.9963	0.9963
S_4	0.9593	0.9576	0.9883	0.9884	**0.9916**	**0.9913**
S_5	0.1562	0.1567	0.9977	0.9975	**0.9980**	**0.9979**
S_6	0.2037	0.2033	0.9983	0.9984	0.9978	0.9976
S_7	0.9329	0.9312	0.9972	0.9973	0.9967	0.9965
S_8	0.9653	0.9654	0.9994	0.9994	0.9981	0.9981
S_9	0.9705	0.9702	0.9975	0.9974	0.9973	0.9971
S_{10}	0.1514	0.1488	0.9992	0.9991	0.9956	0.9952
S_{11}	0.9689	0.9690	0.9974	0.9974	0.9954	0.9953
S_{12}	0.9898	0.9898	0.9975	0.9974	0.9968	0.9967
S_{13}	0.9839	0.9837	0.9945	0.9943	0.9932	0.9930
S_{14}	0.1502	0.1510	0.9996	0.9994	0.9980	0.9981
S_{15}	0.1537	0.1543	0.9981	0.9982	**0.9984**	**0.9982**
avg.	**0.6411**	**0.6407**	0.9972	0.9972	**0.9963**	**0.9962**

Table 9 shows the classification accuracy results of the1D-CNN model when tested on the 3X Unknown scenario using three preprocessing strategies (raw data, normalization, and normalization + PCA). The evaluation includes classifying 7 different classes of MI across 15 subjects using 50 epochs, and the table shows the performance of the model in the training and testing phases for each preprocessing technique.

Table 10 shows the weighted average of the classification performance metrics (Precision, Recall, and F1-Score) for the proposed model using the 3X Unknown scenario with preprocessing techniques for classifying 7 classes.

Table 10. Weighted average of the classification performance metrics (precision, recall, and F1-Score) for Data preprocessing scenarios (7 Classes)

Subjects	Data Preprocessing Scenarios (7 classes)								
	Raw Data			Normalization			Normalization + PCA		
	Pr.	Re.	F1.	Pr.	Re.	F1.	Pr.	Re.	F1.
S_1	0.99	0.99	0.99	1.00	1.00	1.00	0.99	0.99	0.99
S_2	0.08	0.15	0.04	1.00	1.00	1.00	1.00	1.00	1.00
S_3	0.89	0.89	0.89	1.00	1.00	1.00	1.00	1.00	1.00
S_4	0.96	0.96	0.96	0.99	0.99	0.99	0.99	0.99	0.99
S_5	0.22	0.16	0.07	1.00	1.00	1.00	1.00	1.00	1.00
S_6	0.13	0.20	0.14	1.00	1.00	1.00	1.00	1.00	1.00
S_7	0.93	0.93	0.93	1.00	1.00	1.00	1.00	1.00	1.00
S_8	0.97	0.97	0.97	1.00	1.00	1.00	1.00	1.00	1.00
S_9	0.97	0.97	0.97	1.00	1.00	1.00	1.00	1.00	1.00
S_{10}	0.15	0.15	0.05	1.00	1.00	1.00	1.00	1.00	1.00
S_{11}	0.97	0.97	0.97	1.00	1.00	1.00	1.00	1.00	1.00
S_{12}	0.99	0.99	0.99	1.00	1.00	1.00	1.00	1.00	1.00
S_{13}	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99
S_{14}	0.16	0.15	0.05	1.00	1.00	1.00	1.00	1.00	1.00
S_{15}	0.19	0.15	0.09	1.00	1.00	1.00	1.00	1.00	1.00
avg.	0.6387	0.6407	0.606	**0.9987**	**0.9987**	**0.9987**	**0.998**	**0.998**	**0.998**

5. Discussion

The proposed model was trained and tested across different data balance scenarios for imaginary movement classification using the upper limb movement dataset. The training and testing were conducted on only one subject (S_1) for eight classes in order to evaluate the impact of data distribution on the model's accuracy and performance. Observed in Table 5, the scenario performance for No Balance shows the lowest accuracy among all scenarios at 50 epochs, with an accuracy of 0.6765, where notice that the confusion matrix shows that the model classified most of the samples as belonging to the unknown class, regardless of the actual class of the samples. This performance reflects the effect of the lack of balance between classes, which led to a weakness in the model's ability to classify movements correctly. Table 6 shows that the scenario clearly shows a weakness in the performance model. This model was unable to classify the seven basic classes, and (Pr., Re., and F1) were zero for these classes. In contrast, the unknown class performed well due to its numerical dominance, with a precision of 0.68, Recall of 1.00, and F1-score of 0.81. This scenario reflects the negative impact of the unbalanced data distribution on the model's performance.

Notice in Table 5 that the scenario performance for Full Balance shows the best performance among all scenarios at 50 epochs, with accuracy reaching 0.8539. This performance reflects the effect of a balanced distribution for all classes, which enhanced the model's ability to classify different movements with high accuracy. It was observed that the confusion matrix shows the model's consistent performance with a clear decrease in errors. This model was able to classify all classes accurately, as the equal distribution of samples shows a positive effect on reducing confusion between classes. Through Table 6, the model performance in this scenario improved significantly compared to the No balance scenario. All classes showed relatively high values in (Pr., Re., and F1.), indicating that data balance helps improve performance. For example, the EF class performed strongly, While the unknown class recorded average performance compared to the rest of the classes (Precision of 0.67, Recall of 0.01, And F1-Score of 0.03). These results indicate that the model was accurate in predicting the unknown class, but its ability to capture all its samples was weak. In general, the scenario shows great effectiveness in improving the overall performance of the model. Still, it may need to enhance the performance of the unknown class because it is important in practical application.

Now, we observed in Table 5 that scenario performance for 3X Unknown shows with accuracy reaching 0.8467 at 50 epochs, reflecting a performance very close to the Full balanced scenario. Through Fig. 9, the confusion matrix shows a consistent and strong performance of the model, as the classification of the unknown class was improved while reducing the errors in classifying the other classes. This scenario reflects the improvement of the comprehensiveness of the model by increasing the samples of the unknown class, which helped the model to accurately recognize this class alongside classifying the other classes. We notice in Fig. 10 that the performance curves show a gradual stabilization in accuracy and loss with continuous improvement during the training and testing process. The training process was stopped at 78 epochs using early stopping after reaching a clear stabilization without further performance improvement. This scenario performed similarly to the Full balance scenario. The base classes performed well with clear stability, while the Unknown class was more balanced compared to the full balance scenario. This scenario highlights the model's ability to handle balanced data while improving the unknown class's assimilating, as shown in Table 6.

Following in Table 5, scenario performance for 50% Unknown had the lowest performance among all scenarios, with an accuracy of 0.5029 at 48 epochs, where the training process was stopped using early stopping. Although the number of samples for the unknown class was 7 times the minimum for the rest of the classes, this distribution was not sufficient to enable the model to achieve good performance in classifying classes, where notice that the confusion matrix shows that the model struggles to correctly classify the unknown class, with frequent errors across all classes. Table 6 shows that this scenario showed very poor performance compared to the other scenarios. Many classes had very low of (Pr., Re., and F1.), indicating difficulty distinguishing between classes due to an inadequate distribution. The unknown class was the only one that achieved relatively acceptable performance due to the large number of samples. Still, it was not enough to improve the overall performance of the model.

Finally, notice in Table 5 that the scenario performance for Without Unknown (7 Classes) achieved the highest performance among all scenarios, with accuracy reaching 0.9924 at 50 epochs. This outstanding performance reflects the effect of removing the unknown class, which reduced the complexity of the model and contributed to improving its ability to distinguish between the other seven classes. Whereas the confusion matrix shows the model's excellent performance, as most samples are accurately classified into the correct classes with very small errors. In Table 6, the model in this scenario showed almost perfect performance, as all seven basic classes achieved a very high level of (Pr., Re., and F1.). Based on the analysis of the results, stable performance was achieved in two main scenarios, "Full Balance" and "3X Unknown". These scenarios showed a clear balance between all classes, depending on the requirements of the study; the 3X Unknown scenario was chosen as it achieves a balance between high accuracy and balanced performance across all classes, where the accuracy and loss curves reflected stability and convergence between the model's performance on the training and test data, indicating the model's efficiency in dealing with balanced data.

Hyperparameter tuning of the model plays an important role in ensuring the optimal performance of the 1D-CNN network for classifying MI. The hyperparameters were adjusted based on the 3X Unknown data balance scenario to improve the accuracy and stability of the model. As we can notice in Table 4, the effect of the different parameters that were applied is as follows: Case 1 was adopted as the final model setting, where the Adam algorithm was used with a learning rate of 0.001 and a batch size of 224. These settings showed balanced and stable performance, as the training and testing accuracy reached 0.84 for each. This reflects the ability of the low learning rate to gradually improve the model weights while reducing the risk of instabilities during the training process. At the same time, the larger batch size contributed to enhancing the stability of the training. In contrast, using a higher learning rate (0.01) as in the second and fourth cases led to a significant decrease in accuracy. The model could not converge sufficiently due to the large weight updates. Also, the AdamW algorithm used in cases 3 and 4 performed similarly to case 1 at a low learning rate but took longer to train. Case 1 was ultimately chosen because it achieves the ideal balance between high performance and time efficiency, making it the most suitable for achieving the goals of this work.

After selecting the 3X Unknown balance scenario as the best scenario for data balance based on its high accuracy and large number of samples in the unknown class, the model was tested on preprocessing scenarios using the 8-class and 7-class classification types, which were applied to a Subject-by-Subject training scenario. The preprocessing scenarios included using the normalization technique to standardize the data and improve its distribution, in addition to applying the PCA technique to reduce the dimensions to 30 principal components. The experiments were performed using only 50 epochs. The results are shown in Tables 7, 8, 9, and 10.

In Tables 7 and 8, we observed the model's accuracy and values (Pr., Re., and F1.) during the training and testing phases of the 8 classes using the three preprocessing scenarios. The results showed the lowest values of accuracy and (Pr., Re., and F1.) in the Raw Data scenario, with accuracy (ACC) ranging between (0.30 - 0.84) across all subjects. The overall average was low for both the training and testing phases, reflecting the model's poor performance in this scenario. Whereas the accuracy improved significantly using the Normalization scenario compared to the Raw data, with values approaching 0.90 for most subjects and high values for (Pr., Re., and F1.). The overall average showed a significant improvement in this scenario. Finally, the model achieved the best performance using the Normalization + PCA scenario. All subjects showed high accuracy, exceeding the average of 0.91 for both the training and testing phases, the best results for (Pr., Re., and F1.) with average values exceeding 0.90 for most metrics. This reflects the effectiveness of the approach in significantly improving classification accuracy, making it the preferred choice among preprocessing scenarios.

Tables 9 and 10 show the model accuracy and values (Pr., Re., and F1.) during the training and testing phases for 7 Classes using the three preprocessing scenarios. The results showed poor overall performance when using the Raw Data scenario, with accuracy being relatively low in several subjects, such as subject 2 (S_2). The overall average of accuracy and values (Pr., Re., and F1.) was lower than the rest of the preprocessing scenarios. Whereas the accuracy and values (Pr., Re., and F1.) using the Normalization scenario significantly improved the classification of classes compared to Raw data, with almost all subjects achieving very high accuracy and values (Pr., Re., and F1.) exceeding 0.99, indicating stable performance with the exclusion of the unknown class. Finally, the model achieved advanced performance using the Normalization + PCA scenario, with values very close to the normalization scenario, approaching 0.99 in all metrics and for all subjects.

The results showed that preprocessing scenarios significantly impacted the model’s performance in class classification, highlighting the importance of improving data quality through preprocessing. The Normalization and Normalization + PCA scenarios contributed significantly to improving classification accuracy, with the results showing a significant superiority over Raw data. The Normalization + PCA scenario was the best choice among all preprocessing scenarios, achieving the highest levels of accuracy and (Pr., Re., and F1.) across most topics. This strong performance reflects the effectiveness of this scenario in improving class classification when using 8-class classification, which is the focus of this study. These results emphasize the importance of choosing appropriate preprocessing techniques to improve model performance and highlight the effectiveness of the 3X Unknown balance scenario that was chosen. It contributed to achieving a balance between model accuracy and the large number of samples for the unknown class.

After analyzing the performance of the proposed advanced model on preprocessing scenarios, we will highlight some previous works in terms of model performance, number of channels, etc., to evaluate the progress of this study, as shown in Table 11.

First, number of channels used: Most previous studies, such as [11] and [16], relied on a small number of channels, as some studies used only 4 and 11 channels, which reduces the complexity of the data but may affect the accuracy of prediction. In contrast, this study used 61 EEG channels, which helped improve the discrimination between different classes of MI and achieved higher classification accuracy compared to studies that relied on a smaller number of channels. Secondly, preprocessing techniques: Previous studies focused mainly on using bandpass filters as one of the basic processing methods. In contrast, the current study adopted advanced processing techniques such as Z-Score Normalization and PCA, which reduced the dimensions and improved the accuracy of the model, making it more efficient compared to traditional methods. Thirdly, classification type: Some previous studies, such as [12] and [14], focused on binary classification, while some other studies focused on classifying a limited number of classes, such as [13] and [15], which relied on classifying only 6 classes. In contrast, this study adopted multi-class classification with eight classes, including the unknown class, which enhances the model’s ability to deal with more complex scenarios and provides a broader range of potential applications. Fourth, classification algorithm: Some previous studies used traditional algorithms such as SLDA and RLDA, as in [14] and [11], while other studies adopted CNN-based networks such as TSPNet, TTSN, and TDLNet as in [13,15, and 16]. This study adopted a 1D-CNN, which has proven effective in classifying MI, with less computational resource consumption than more complex deep networks. Finally, model performance: The results showed that the proposed model outperformed previous studies, achieving high accuracy when classifying MI, which can be observed in Table 11, which reflects the efficiency of the adopted methodology in improving performance by using a large number of channels, adopting more advanced processing techniques, and supporting multi-class classification.

Received: December 28, 2024. Revised: March 8, 2025.

Table 11. Summary of related work and proposed model for BNCI Horizon 2020 dataset

Ref & year	Task Type	Preprocessing	Feature Extraction	Classifier	Training Scenario	No. Channels Used	Accuracy (%)
[11] 2020	EF, EE, FS, FP, HO, HC, and Rest	Interpolation and an antialiasing filter, artifact removal, large Laplacian spatial filter, and channel reduction	SSSF & mean amplitude	RLDA	Subject-by-Subject	4 [C1, C2, CPz, Cz] for MRCP activation	**Binary/** MI: Avg. 73 (mov vs. rest), **Multi-class (7-class):** Avg. 38
[12] 2020	EF, EE, FS, FP, HO, HC, and Rest	Bandpass, Segmentation, and Region of interest-ROI	Continuous Wavelet Transform, and TF Maps	DCNN	Subject-by-Subject	61	**Binary**/MI: **ACC.:** Avg. 62.47 (premov vs. premov), and Avg. 90.3 (premov vs. rest), **F-score:** Avg. 57.70 (premov vs. premov), and Avg. 88.79 (premov vs. rest)
[13] 2023	EF, EE, FS, FP, HO, HC	Bandpass Filter, Notch Filter, and Down sample	TDFE, SDFE, and TSPFE	TSPNet (CNN - based)	Subject-by-Subject	61	**Multi-class (6-class)** ACC: Avg. 49.7
[14] 2023	EF, EE, Rest	Down sample, Bandpass Filter, Divide data into time periods, and ICA	Amplitude, and Entropy features (Rényi and Shannon)	sLDA	Subject-by-Subject	31	**Binary** MI: Acc, and F-score: max. 95 (mov vs. rest) Acc,: max. 53, and F-score: max. 55 (mov vs. mov)
[15] 2023	EF, EE, FS, FP, HO, HC	Down sample	TDM, Inception Module, and RAMM	TDLNet (CNN - based)	All Subjects Together	61	**Multi-class (6-class):** Acc.: 63
[16] 2024	EF, EE, FS, FP, HO, HC, and Rest	Down sample, Bandpass Filter, Divide data into time periods, and ICA, Normalization, Movement onset localization	CNN+TRCA	TTSNet (CNN - based)	/	11	**Binary** /MI: Acc Precise onset-labeled/dataset I(a): 0.7707 Unlabeled onset/dataset I(b): 0.7526 **Multi-class**/ MI: Acc Precise onset-labeled/dataset I(a): 0.4588 Unlabeled onset/dataset I(b): 0.4141
Proposed model	EF, EE, FS, FP, HO, HC, Rest, and unknown	Z-Score Normalization, and PCA	/	1D-CNN	Subject-by-Subject	61	**Multiclass** MI: ACC.: **8-class**/ Training: Avg. 91.01, Testing: Avg. 90.99, F1-score: Avg.90.6 **7-class**/ Training: Avg. 99.63, Testing: Avg. 99.62, F1-score: Avg. 99.8

Received: December 28, 2024. Revised: March 8, 2025.

6. Conclusion

In this study, the 1D-CNN model was customized and tested to classify imaginary movements using EEG signals from the 2020 BNCI Horizon database. The results showed that the model is able to achieve high classification accuracy in different training and testing scenarios, with the 3X Unknown scenario being the most suitable for the study due to its high accuracy and balance in dealing with the unknown class. The study used preprocessing techniques, such as Normalization + PCA, which contributed to improving the model's performance and reducing the data complexity.

However, the study revealed some challenges that need to be addressed in the future, such as adopting a subject-dependent approach, which limits the generalization of the model to new users. The study also relied on a pre-recorded database dedicated to healthy individuals, which may reduce the model's ability to adapt to the needs of individuals with special health conditions such as cerebral palsy or amputation.

In the future, this work can be expanded by exploring the use of other deep learning models, such as RNN, for more complex analysis of EEG signals. In addition, it is proposed to adopt a training scenario based on collecting data from all subjects instead of subject-by-subject, which contributes to producing a more general and flexible model. Also, it is proposed that transfer learning be adopted to enhance adaptability across individuals. Finally, the model can be tested in realistic operating environments and connected to practical devices such as prosthetic limbs to evaluate its performance in practice and improve its real-world applications

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Alazzawi has done conceptualization, supervision, and project administration. I. AL-Qinani has done Methodology, software, validation, formal analysis, data curation, visualization, and writing—original draft preparation. Investigation, writing—review and draft preparation. Investigation, writing—review and editing, and resources have been done by Y. Alazzawi and I. AL-Qinani.

Acknowledgments

The authors would like to thank the Informatics Institute for Postgraduate Studies, Mustansiriyah University (www. uomustansiriyah.edu.iq), and Baghdad University, Baghdad-Iraq, for their support in the present work.

References

[1] V. Jayashekar, and R. Pandian, "Hybrid Feature Extraction for EEG Motor Imagery Classification Using Multi- Class SVM", *International Journal of Intelligent Engineering and Systems*, Vol. 15, No. 4, pp. 20-30, 2022, doi: 10.22266/ijies2022.0831.03.

[2] J-H. Jeong, J-H. Cho, Y-E. Lee, S-H. Lee, G-H. Shin, Y-S. Kweon, JdR. Millán, K-R. Müller, and S-W. Lee, "2020, International Brain–Computer Interface Competition: A review", *Frontiers in Human Neuroscience*, Vol. 16, No. 898300, pp. 1-23, 2022.

[3] G. Zhan, S. Chen, Y. Ji, Y. Xu, Z. Song, J. Wang, L. Niu, J. Bin, X. Kang, and J. Jia, "EEG-Based Brain Network Analysis of Chronic Stroke Patients After BCI Rehabilitation Training", *Frontiers in Human Neuroscience*, Vol. 16, No. 909610, pp. 1-16, 2022.

[4] O. Amanuel, and Y. Alazzawi, "BCI-Based Smart Room Control using EEG Signals", *Al-Khwarizmi Engineering Journal*, Vol. 18, No. 4, pp. 60 -72, 2022.

[5] D. Yadav, S. Yadav, and K. Veer, "A comprehensive Assessment of Brain Computer Interfaces: Recent Trends and Challenges", *Journal of Neuroscience Methods*, Vol. 346, No. 108918, pp. 1-20, 2020.

[6] O. Amanuel, and Y. Alazzawi, "Design and Implementation of EEG-Based Smart Structure", *International Journal of Intelligent Engineering and Systems*, Vol. 16, No. 1, pp. 314-327, 2023, doi: 10.22266/ijies2023.0228.28.

[7] M. J. N. Francis, M. P. Keran, R. Chetan, and B. N. Krupa, "EEG-Controlled Robot Navigation using Hjorth Parameters and Welch-PSD", *International Journal of Intelligent Engineering and Systems*, Vol. 14, No. 4, pp. 231-240, 2021, doi: 10.22266/ijies2021.0831.21.

[8] A. Wijaya, T. B. Adji, and N. A. Setiawan, "Logistic Regression based Feature Selection and Two-Stage Detection for EEG based Motor Imagery Classification", *International Journal of Intelligent Engineering and Systems*, Vol. 14, No. 1, pp. 134 -146, 2021, doi: 10.22266/ijies2021.0228.14.

[9] M. H Kabir, N. I. Akhtar, N. Tasnim, A. S. M. Miah, H.-S. Lee, S.-W. Jang, and J. Shin, "Exploring Feature Selection and Classification Techniques to Improve the Performance of an

Electroencephalography-Based Motor Imagery Brain–Computer Interface System”, *Sensors*, Vol. 24, No. 4989, pp. 1 -27, 2024.

[10] BNCI Horizon 2020, http://bnci-horizon-2020.eu/,“ Data sets /25. Upper limb movement decoding from EEG (001-2017)” 2017. [Online]. Available: http://bnci-horizon-2020.eu/database/data-sets. [Accessed: Nov. 29,2023].

[11] J.-H. Jeong, N.-S. Kwak, C.Guan, and S.-W. Lee, “Decoding Movement-Related Cortical Potentials Based on Subject-Dependent and Section-Wise Spectral Filtering”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol. 28, No. 3, pp. 687-698, 2020.

[12] N. Mammone, C. Ieracitano, and F. C. Morabito, “A Deep CNN Approach to Decode Motor Preparation of Upper Limbs from Time-Frequency Maps of EEG Signals at Source Level”, *Neural Networks*, Vol. 124, No. 2020, pp. 357–372, 2020

[13] J. Bi, M. Chu, G. Wang, and X. Gao, “TSPNet: A Time-Spatial Parallel Network for Classification of EEG-Based Multiclass Upper Limb Motor Imagery BCI”, *Frontiers in Neuroscience.*, Vol. 17, No. 1303242, pp. 1-14, 2023.

[14] L. Batistić, J. Lerga, and I. Stanković, “Detection of Motor Imagery Based on Short-Term Entropy of Time–Frequency Representations”, *BioMedical Engineering OnLine*, Vol. 22, No. 41, pp. 1-23, 2023.

[15] J. Bi, and M. Chu, “TDLNet: Transfer Data Learning Network for Cross-Subject Classification Based on Multiclass Upper Limb Motor Imagery EEG”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol. 2023, No. 21, pp. 3958-3967, 2023.

[16] H. Jia, S. Han, C. F. Caiafa, F. Duan, Y. Zhang, Z. Sun, and J. Solé-Casals, “Enabling Temporal–Spectral Decoding in Multi-Class Single-Side Upper Limb Classification”, *Engineering Applications of Artificial Intelligence*, Vol. 133, Part E, No. 108473, pp. 1-12, 2024.

[17] P. Ofner, A. Schwarz, J. Pereira, and G. R. Muller-Putz, “Upper Limb Movements Can Be Decoded from the Time-Domain of Low-Frequency EEG”, *PLoS One*, Vol. 12, No. 8, pp. 1-24, 2017.

[18] I. Rakhmatulin, M.-S. Dao, A. Nassibi, and D. Mandic, “Exploring Convolutional Neural Network Architectures for EEG Feature Extraction”, *Sensors*, Vol. 24, No. 877, pp. 1-39, 2024.

[19] R. Sánchez-Reolid, F. L. d. Rosa, M.T. López, and A. Fernández-Caballero, “One-dimensional Convolutional Neural Networks for Low/High Arousal Classification from Electrodermal Activity”, *Biomedical Signal Processing and Control*, Vol. 71, No. 103203, pp. 1-9, 2022.

[20] S. Chaabene, B. Bouaziz, A. Boudaya, A. Hökelmann, A. Ammar, and L. Chaari, “Convolutional Neural Network for Drowsiness Detection Using EEG Signals”, *Sensors*, Vol. 21, No. 1734, pp. 1-19, 2021.

[21] V. Safavi, A. M. Vaniar, N. Bazmohammadi, J. C. Vasquez, O. Keysan, and J. M. Guerrero, “Early Prediction of Battery Remaining Useful Life Using CNN-XGBoost Model and Coati Optimization Algorithm”, *Journal of Energy Storage*, Vol. 98, No. 113176, pp. 1-12, 2024.

[22] M. Saini, U. Satija, and M. D. Upadhayay, “Light-Weight 1-D Convolutional Neural Network Architecture for Mental Task Identification and Classification Based on Single-Channel EEG”, *arXiv preprint arXiv:2012.06782*, pp. 1-11, 2020.

[23] A. Apicella, F. Isgrò, A. Pollastro, and R. Prevete, “On the Effects of Data Normalization for Domain Adaptation on EEG Data”, *Engineering Applications of Artificial Intelligence*, Vol. 123, No. 106205, pp. 1-14, 2023.

[24] X. Wang, V. Liesaputra, Z. Liu, Y. Wang, and Z. Huang, “An In-Depth Survey on Deep Learning-Based Motor Imagery Electroencephalogram (EEG) Classification”, *Artificial Intelligence in Medicine*, Vol. 147, No. 102738, pp. 1-20, 2024.

[25] L. Peng, Z. Lu, T. Lei, and P. Jiang, “Dual-Structure Elements Morphological Filtering and Local Z-Score Normalization for Infrared Small Target Detection Against Heavy Clouds”, *Remote Sensing.*, Vol. 16, No. 2343, pp. 1 -29, 2024.

[26] W. Jia, M. Sun, J. Lian, and S. Hou, “Feature Dimensionality Reduction: A Review”, *Complex & Intelligent Systems*, Vol. 8, pp. 2663–2693, 2022.

[27] Z. Huang, and M. Wang, “A Review of Electroencephalogram Signal Processing Methods for Brain-Controlled Robots”, *Cognitive Robotics*, Vol. 1, pp. 111-124, 2021.

[28] N. T. Mahmood, M. H. Al-Muifraje, and S. K. Salih, “EMG Signals Classification of Wide Range Motion Signals for Prosthetic Hand Control”, *International Journal of Intelligent*

Engineering and Systems, Vol. 14, No. 5, pp. 410- 421, 2021, doi: 10.22266/ijies2021.1031.36.

[29] G. Zhang, C. D. Carrasco, K. Winsler, B. Bahle, F. Cong, and S. J. Luck, "Assessing the Effectiveness of Spatial PCA on SVM-Based Decoding of EEG Data", *NeuroImage*, Vol. 293, No. 120625, pp. 1-12, 2024.

[30] D. Pawar, and S. N. Dhage, "Feature Extraction Methods for Electroencephalography Based Brain-Computer Interface: A Review", *IAENG International Journal of Computer Science*, Vol. 47, No. 3, pp. 1-115, 2020.

[31] M. B. G. Gowda, N. K. Boraiah, V. Eshappa, and G. Chandrashekara, "Classification of Epileptic EEG Signals Using Improved Atomic Search Optimization Algorithm", *International Journal of Intelligent Engineering and Systems*, Vol. 16, No. 6, pp. 134-144, 2023, doi: 10.22266/ijies2023.1231.12.

[32] S. K. Jarallah, and S. A. Mahmood, "Query-Based Video Summarization System Based on Light Weight Deep Learning Model", *International Journal of Intelligent Engineering and Systems*, Vol. 15, No. 6, pp. 247-262, 2022, doi: 10.22266/ijies2022.0831.57.

[33] A. D. Wibawa, E. S. Pane, D. Risqiwati, and M. H. Purnomo, "Rules Extraction of Relevance Vector Machine for Predicting Negative Emotions from EEG Signals", *International Journal of Intelligent Engineering and Systems*, Vol. 15, No. 1, pp. 42-54, 2022, doi: 10.22266/ijies2022.0228.05.

[34] A. W. Widodo, S. Handoyo, I. Rupiwardani, Y. T. Mursityo, I. N. Purwanto, and H. Kusdarwati, "The Performance Comparison between C4.5 Tree and One-Dimensional Convolutional Neural Networks (CNN1D) with Tuning Hyperparameters for the Classification of Imbalanced Medical Data", *International Journal of Intelligent Engineering and Systems*, Vol. 16, No. 5, pp. 748-759, 2023, doi: 10.22266/ijies2023.1031.63.