From Thought to Motion: A 1D-CNN Approach to Comparing Real and Imagined Activities

Akib Sadmanee*
group 3
sadmanee@hawaii.edu

Amila Indika*
group 3
amilaind@hawaii.edu

Md Rahat Shahriar Zawad*
group 3
zawad@hawaii.edu

SK Fazlee Rabby*
group 3
sfrabby@hawaii.edu

***Abstract*—This study investigates the capability of 1D Convolutional Neural Networks (CNNs) to differentiate between real and imagined motor movements using electroencephalography (EEG) signals with minimal preprocessing. Two hypotheses were tested: the first hypothesis posited that the CNN model could distinguish between real and imagined right and left fist movements, while the second hypothesis extended the model's application to both fists and feet. The results indicate high classification performance for both hypotheses. For Hypothesis 1, the model achieved an overall accuracy of 96.07% and an F1-score of 0.96 across various classes, with the loss recorded at 0.1251. For Hypothesis 2, the model demonstrated an overall accuracy of 96.90% and an F1-score of 0.97, with a loss of 0.1519. These outcomes suggest that 1D CNNs are effective in decoding EEG signals for both types of motor movements, highlighting their potential in applications such as brain-computer interfaces and neurorehabilitation tools.**

***Index Terms*—Brain-Computer Interface, Deep Neural Networks, Convolutional Neural Networks, EEG Data**

I. Introduction

Brain-computer interfaces (BCIs) facilitate direct communication between neural activity and external apparatus through a sophisticated system comprising signal acquisition, decoding methodologies, and device actuation mechanisms [1]. These systems are instrumental in addressing cognitive and sensory-motor deficits and enhancing recreational technologies [2]–[5]. The architecture of BCIs integrates devices for neural signal detection, which capture user intent, algorithms to interpret these signals, and actuators to manage external devices such as computers and prosthetic limbs [1].

Various studies have explored invasive and non-invasive neural recording techniques to gauge user intentions via brain activity effectively. Invasive methods, while offering high accuracy, carry substantial risks associated with surgical implantations, such as those involving micro-electrode arrays [2], [6]–[9]. Conversely, non-invasive strategies like EEG, fMRI, and fNIRS present safer, user-friendly alternatives that avoid surgical interventions [10], [11]. Particularly, EEG-based techniques have gained popularity due to the increasing availability of commercial devices, offering cost-effective solutions for BCI development [12], [13].

EEG signal analysis employs paradigms such as visual evoked potentials (VEP) and motor imagery (MI) to decode user intentions [2], [3], [14], [15]. VEP, for instance, involves recording electric potentials triggered by visual stimuli, although this method is susceptible to external disruptions [16]–[18]. Conversely, MI involves imagined movements and induces distinctive neural patterns in the sensorimotor cortex, providing a versatile basis for application across various activities [19], [20][19, 20]. However, significant challenges like high variability and low signal-to-noise ratios in EEG signals necessitate complex preprocessing to mitigate artifacts and enhance data clarity [21], [22].

Recent advancements in signal processing techniques such as independent component analysis (ICA), wavelet transform (WT), and empirical mode decomposition (EMD) have facilitated more effective EEG data preprocessing, albeit requiring extensive expert intervention [23]–[31]. These preparatory steps are essential for accurate feature extraction and classification, which are critical for the functioning of BCIs [32], [33].

Deep learning (DL) approaches have revolutionized EEG-based BCIs, improving feature extraction and classification tasks [1]. Techniques such as convolutional neural networks (CNNs), inspired by the brain's visual processing architecture, utilize layered structures and non-linear processing units to enhance computational efficiency and reduce parameter counts [34]–[37]. Using CNNs in MI-based EEG classification has shown promising results, allowing for effective pattern recognition and signal classification without requiring intensive preprocessing [1], [38]–[41].

Research indicates that CNNs achieve significant accuracy in classifying MI tasks within EEG datasets, as evidenced by studies conducted by Dose et al. and Tang et al., who reported classification accuracies of 65.73% and 85.83%, respectively [29], [42]. More recent innovations, such as those proposed by Lun et al., demonstrated even higher accuracies (97.28%) by optimizing electrode configurations and introducing novel network feeding strategies focusing on temporal features using 1D convolutional layers [43][43]. These developments underscore the potential of CNNs in enhancing the efficacy and accessibility of EEG-based BCI systems.

This report is organized into seven main sections, aiming to explore using 1D convolutional neural networks (1D-CNN) to analyze brain activity from real and imagined motor tasks. Section 3, Related Work, reviews existing studies on neural decoding and advancements in CNNs applied to EEG data.

*All the authors have equal contributions. The order was selected alphabetically.

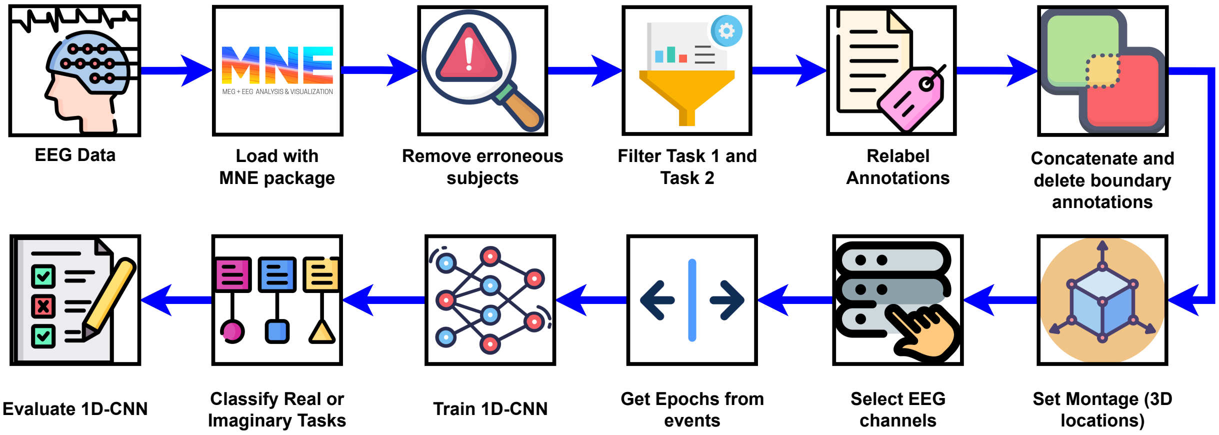
EEG Data
Load with MNE package
MNE
MEG + EEG ANALYSIS & VISUALIZATION
Remove erroneous subjects
Filter Task 1 and Task 2
Relabel Annotations
Concatenate and delete boundary annotations
Set Montage (3D locations)
Select EEG channels
Get Epochs from events
Train 1D-CNN
Classify Real or Imaginary Tasks
Evaluate 1D-CNN

Fig. 1. The proposed 1D-CNN architecture overview for classifying real and imaginary motor imagery tasks

Section 4, Methods, details our methodological approach, including problem definition, proposed solution, dataset characteristics, data preprocessing, and the specific 1D-CNN architecture used. Section 5, Results and Evaluation, discusses the effectiveness of our model in distinguishing between EEG patterns of real and imagined movements. Section 6, Discussion, interprets the results, considers the implications for brain-computer interfaces, addresses study limitations, and suggests future research directions. Section 7, Conclusion, summarizes our findings and their relevance to computational neuroscience and neural engineering while reflecting on this research's impact and future prospects.

II. Related Work

Motor imagery (MI) decoding from electroencephalography (EEG) data is a crucial aspect of brain-computer interface (BCI) systems, enabling communication and control applications for individuals. Various studies have addressed this challenge and developed methodologies and neural network architectures to improve classification accuracy and robustness across different subjects and tasks.

One approach involves ensemble learning techniques coupled with curriculum learning and intra-ensemble distillation to address individual differences in EEG signals. [44][1] This method surpasses existing techniques by leveraging a two-stage model ensemble architecture with multiple feature extractors and a shared classifier. This approach exhibits high learning capacity and robust performance by incorporating novel loss terms and ensemble strategies, promising advancements in calibration-free BCIs, and addressing domain shift challenges in multi-subject EEG datasets.

Another strategy focuses on developing specialized neural network architectures tailored for cross-subject EEG motor imagery classification. [45][2] For instance, Karel et al. propose EEGNet Fusion, a multi-branch 2D convolutional neural network (CNN) that adapts to different subjects' data through varied hyperparameter values for each branch. While achieving high accuracy in classifying motor actions across subjects, this model incurs higher computational costs than alternative approaches.

Besides, researchers have explored innovative techniques, such as transforming EEG signals into images for classification using deep learning architectures. [46][3]. Fadel and his team have introduced a model combining deep convolutional neural networks (DCNNs) for spatial and frequency feature extraction with Long Short-Term Memory (LSTM) networks for temporal feature extraction. By representing EEG signals as images and leveraging deep learning, this approach demonstrates promising classification accuracies, outperforming traditional machine learning methods like Support Vector Machines (SVM).

In their research, Kollod et al. compared the performance of various neural network architectures and evaluated the efficacy of transfer learning in enhancing classification accuracy for MI EEG signals [47] [4]. They compare established neural networks for MI EEG signal classification, including Shallow ConvNet and Deep ConvNet. Another study demonstrates the effectiveness of transfer learning in improving classification accuracy, showcasing the potential of pre-trained models like SqueezeNet for EEG MI signal classification tasks. [48][5]

In a recent study, novel approaches like utilizing one-dimensional convolutional neural networks (1D-CNNs) combined with data augmentation and transfer learning have shown promise in achieving high classification accuracy and individualized training for MI EEG-based BCIs. [49][6] The study introduces a 10-layer 1D-CNN architecture capable of achieving high accuracy levels and offers potential for future BCI applications, particularly with portable recording devices and individualized training paradigms.

Most studies utilizing the Physionet Dataset have primarily aimed to tackle issues arising from individual variations and enhance predictive accuracy for various experimental tasks. Although all these researches show a promising future direction in the research of BCIs, there has been minimal exploration into whether this dataset can differentiate between tasks performed in reality versus imagination. This research focuses on understanding how EEG data can contribute to differentiating imaginary and real tasks at hand using machine learning techniques. Such research could pave the way for further exploration into mental disorders associated with hallucinations and imagination.

III. Methods

A. Problem Definition

The primary aim of this study is to differentiate between real and imaginary motor imagery tasks utilizing 1D convolutional neural networks. Consequently, we have devised the following hypotheses to tackle this issue.

> **Hypothesis 1**: The EEG signals for Real and Imaginary motions of Right and Left fists can be distinguished using 1D CNNs with minimal preprocessing and alteration of the data.

> **Hypothesis 2**: The EEG signals for Real and Imaginary motions of both fists and both feet can be distinguished using 1D CNNs with minimal preprocessing and alteration of the data.

B. Proposed solution

The illustration in Figure 1 below depicts the proposed solution overview for classifying real or imaginary motor imagery tasks using a 1D-CNN architecture. Subsequent sections provide further details on implementing the 1D-CNN.

C. Dataset

The dataset for this study encompasses extensive EEG recordings acquired from 109 volunteers, employing a high-density 64-channel EEG system, strictly adhering to the International 10-10 electrode placement system, albeit excluding electrodes Nz, F9, F10, FT9, FT10, A1, A2, TP9, TP10, P9, and P10. Each participant engaged in a series of 14 experimental EEG runs, meticulously recorded using the BCI2000 system, a renowned platform in brain-computer interface research [cite]. The recordings include two baseline runs of one minute each—first with eyes open and then closed—followed by three two-minute sessions for each of the four designated tasks. This structured approach resulted in a substantial volume of data, specifically 218 minutes of baseline recordings and 2616 minutes dedicated to task-specific activities, all sampled at 160 samples per second and encapsulated in the EDF+ format.

The dataset details participant engagement in four distinct tasks, each designed to evoke motor and imagery responses, which are integral to BCI applications:

Active Motor Response with Lateral Visual Cue: Subjects respond to visual cues by physically opening and closing the fist on the side of the displayed target until it vanishes. Motor Imagery with Lateral Visual Cue: Participants imagine the movement of opening and closing the fist corresponding to the side of the visual cue without actual physical movement. Active Motor Response with Vertical Visual Cue: On cues appearing at the top or bottom of the display, subjects perform physical opening and closing of both fists (top cue) or both feet (bottom cue). Motor Imagery with Vertical Visual Cue: Subjects imagine the movement of opening and closing both fists or feet in response to the cues at the top or bottom of the display, respectively.

These tasks are designed to simulate both real and imagined movements, testing the subjects' ability to control their neural signals without actual motor output—a fundamental aspect of effective BCI systems.

Each EEG recording is meticulously annotated to mark specific task-related events: T0 indicates rest periods, T1 signals the onset of actual or imagined movement (left fist or both fists), and T2 denotes the initiation of movements for the right fist or both feet. These annotations are crucial for subsequent data analysis, allowing researchers to synchronize neural activity with expected task execution precisely. The EEG signals were recorded with a resolution of 16-bit per sample, ensuring high-fidelity data capture. The dataset's EDF+ format is compatible with a wide range of EEG analysis software, including the PhysioToolkit environment, where .event files corresponding to each recording are available. These files are essential for researchers intending to study the timing and properties of neural responses associated with controlled motor tasks and their imagined counterparts in a BCI context.

D. Data Preprocessing

Initially, we loaded the data, excluding six participants (IDs 38, 88, 89, 92, 100, and 104) due to annotation errors. This refinement reduces the participant count from 109 to 103. The EEG signals in the original dataset are in raw EDF format, and we utilize the Python MNE package [51] for loading and processing these signals. Our analysis compares the disparities between task imagination and actual task execution. Therefore, we concentrate solely on Task 1 (experimental runs 3, 7, 11) for Hypothesis 1, (5, 9, 13) for Hypothesis 2 and Task 2 (experimental runs 4, 8, 12) for Hypothesis 1, (6, 10, 14) for Hypothesisi 2. Subsequently, we truncate experimental runs exceeding 124 seconds to ensure uniform signal length across inputs.

Following data loading and processing, we convert annotations within the raw EDF signals into five classification labels, utilizing the convenient annotation parser provided by MNE. The original annotations consist of three codes: T0 corresponds to rest, T1 signifies the onset of motion (actual or imagined) of the left fist (in runs 3, 4, 7, 8, 11, and 12), and T2 corresponds to the onset of motion (actual or imagined) of the right fist (in runs 3, 4, 7, 8, 11, and 12). However, to align with our task requirements, we reclassify these annotations into five distinct labels for each of the hypotheses as Presented in *TABLE I* and *TABLE II*:

Then, we merge all raw EEG signals into one extended EEG signal using the MNE package's concatenate_raws() function. Following this step, we eliminate any ”BAD boundary” or ”EDGE boundary” annotations resulting from the concatenation of raw signals.

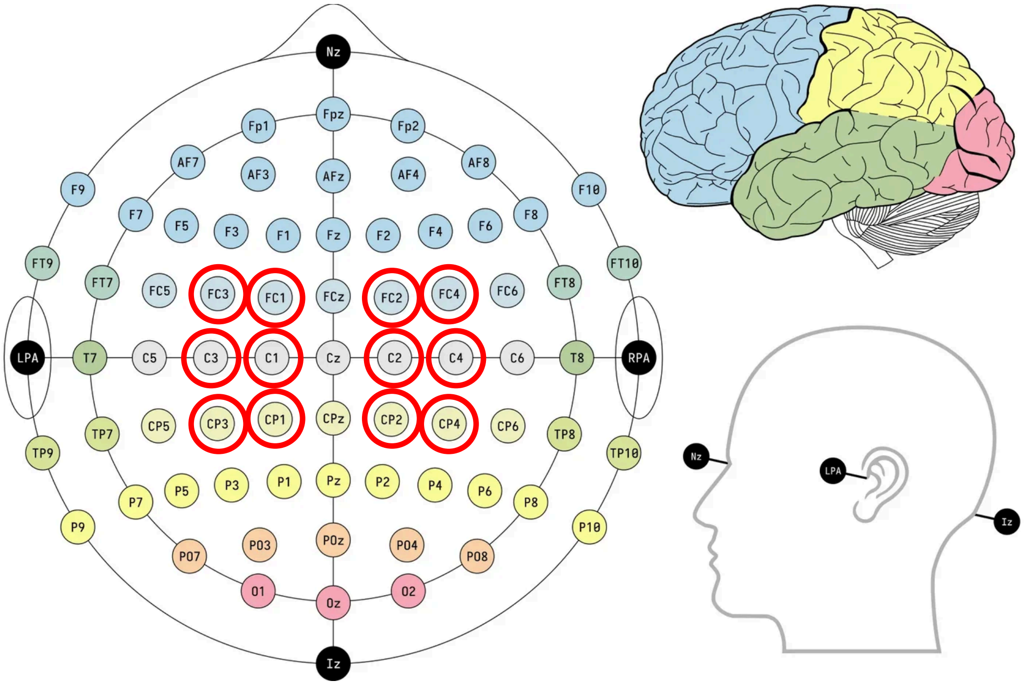

Fig. 2. Twelve channels were selected as the region of interest for the exploration. Adapted from [50]

TABLE I
CLASSES USED IN THE TESTS OF THE MODEL. (FIRST HYPOTHESIS)

Class label	Description of the class
LI	motor imagination of opening and closing the left fist
RI	motor imagination of opening and closing the right fist
LR	actual motor movement of opening and closing the left fist
RR	actual motor movement of opening and closing the right fist
B	resting state (baseline)

TABLE II
CLASSES USED IN THE TESTS OF THE MODEL. (SECOND HYPOTHESIS)

Class label	Description of the class
BI	motor imagination of opening and closing both fists
BR	actual motor movement of opening and closing the right fist
FI	motor imagination of opening and closing both feet
FR	actual motor movement of opening and closing both feet
B	Baseline

Then, we assign the electrode positions in 3D (x, y, z in meters) to preprocessed EEG data using montages. These 3D positions, relative to the brain, are crucial for computing forward solutions and inverse estimates. To achieve this, we utilize the standard 1005 montage provided by MNE. Following this step, we proceed to select channels of interest. According to Mattioli et al. [49], the Region Of Interest (ROI) E demonstrated the best performance for the 1D-CNN model.

Hence, we adopt the channels within ROI E for our study. ROI E encompasses 12 channels: FC1, FC2, FC3, FC4, C1, C2, C3, C4, CP1, CP2, CP3, and CP4 which is illustrated in Figure 2. Then lastly, we segment the data into epochs following the events, each lasting four seconds. These epochs will serve as the input signal for the neural network model.

E. 1D-CNN architecture

The current study introduces a brain-computer interface (BCI) system utilizing a one-dimensional convolutional neural network (1D-CNN). This architecture is distinguished by the convolution process where CNN kernels are confined to slide across a single dimension—time—of the input data. Specifically, the 1D-CNN processes an input matrix sized $M \times N$, where M represents the duration of the time window under analysis, and N denotes the number of electroencephalogram (EEG) channels. In this implementation, M is set to 640, reflecting a temporal window subdivided into 160 steps, each spanning 4-time units, and N is 2, corresponding to two symmetric EEG channels.

Each convolutional layer in this 1D-CNN employs a kernel with dimensions $Q \times N$, where Q captures the extent of the temporal window that the filter covers, maintaining $N = 2$ to ensure the kernel's movement is limited to the temporal dimension without crossing into the channel dimension. This design choice simplifies the convolution operation and focuses on extracting temporal features from the EEG signals, which is crucial for decoding user intentions in BCI applications. The computational model for each layer can be mathematically

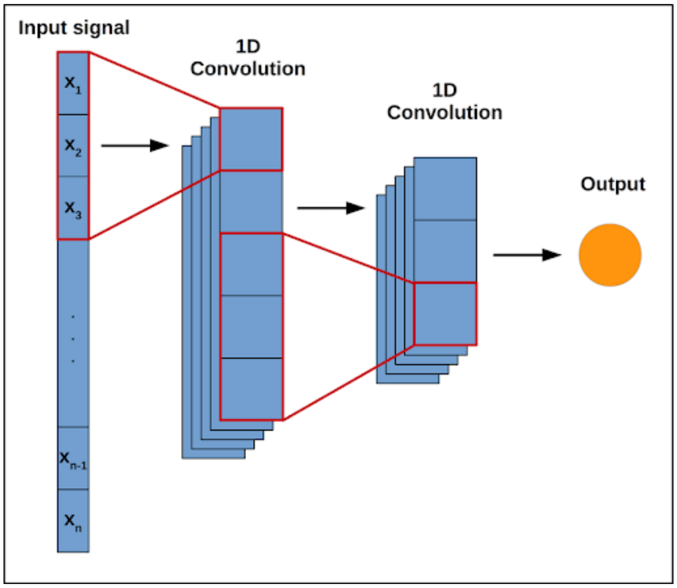

Fig. 3. Sample 1D-CNN architecture with two convolutional layers. [cite]

represented as follows:

$$y_r = f\left(\sum_{q=1}^{Q}\sum_{n=1}^{N} w_{qn}x_{r+q,r+n} + b\right) \tag{1}$$

In this context, y_r represents the output from unit r of the filter feature map, which possesses a dimensionality R. This dimension R equates to M when a stride of 1 is employed along with padding. The variable x denotes the two-dimensional segment of the input that overlaps with the filter, while w and b signify the convolutional filter's connection weights and bias term, respectively. Additionally, f represents the activation function applied within the filter.

To elucidate the computation of the filter feature map's dimensions post-convolution (R), one may utilize the following equation:

$$R = \left\lfloor \frac{M-(K-1)+2\times P}{S} \right\rfloor \tag{2}$$

Here, P represents the padding size, which is defined as the addition of zero-valued pixels around the periphery of the image. This padding technique maintains a consistent input-output dimensional ratio that matches the stride. The stride, denoted as S, refers to the number of pixel positions that the convolutional filter advances at each step during the convolution process. This ensures the filter skips positions systematically, facilitating a structured traversal across the image plane. This formula assists in determining the resultant size of the feature map, thus facilitating a deeper understanding of the convolutional operation's spatial impact on the input data.

The architectural details of the Convolutional Neural Network (CNN) are delineated in Table III. The primary convolutional layer, denoted as L1, employs 32 filters, each with dimensions of 20×20, a stride length of 1, and utilizes padding that maintains the original dimensions of the input image, as indicated by the designation 'SAME' in Table *TABLE III*, consistent with TensorFlow's nomenclature. After L1, a batch normalization (BN) process is implemented, as referenced in *Fig. 3*. Batch normalization standardizes the inputs for the subsequent layer, significantly enhancing learning rates and exerting regularization effects that foster improved model generalization.

Batch normalization operates distinctively during the training and testing phases. It normalizes and centers the inputs around zero during training, utilizing the mean (μ) and variance (σ^2) calculated across the current batch. This batch is the collection of instances from which the loss and gradient are computed for the learning algorithm, enabling the model to adjust optimally to the scaling of the inputs. This normalization strategy, therefore, plays a crucial role in stabilizing the learning process and facilitating faster convergence of the neural network.

$$\mu = \frac{1}{b}\sum_{i=1}^{b} X^{(i)} \tag{3}$$

$$\sigma^2 = \frac{1}{b}\sum_{i=1}^{b}(X^{(i)}-\mu)^2 \tag{4}$$

where b is the number of instances in the batch and $X^{(i)}$ is an instance. Then, the zero-centered normalized value $\hat{X}^{(i)}$ for each instance is computed, where $\xi = 10^{-5}$ is used to avoid zero divisions:

$$\hat{X}^{(i)} = \frac{X^{(i)}-\mu}{\sqrt{\sigma^2+\xi}} \tag{5}$$

In this equation: $\hat{X}^{(i)}$ represents the zero-centered normalized value of the i-th instance, $X^{(i)}$ is the i-th instance of the data, μ is the mean of the data batch, σ^2 is the variance of the data batch, and ξ (a small constant, often used to ensure numerical stability during division by avoiding division by zero) is added to the variance in the denominator. The normalization might not be good for a given task, so BN adds a further step during training, using trained parameters, that further scales and offsets the values as needed:

$$z^i = \gamma \otimes \hat{X}^{(i)} + \beta \tag{6}$$

In the described architecture, the operation denoted by $\otimes$ represents the element-wise product between each input value and a corresponding scale parameter, γ. These scale parameters and offset parameters β are optimized during the learning process through backpropagation. In the test phase, the batch-specific mean (μ) and variance (σ^2) are not directly calculable, necessitating moving averages of these parameters, previously computed during the training phase, to ensure model consistency and performance.

The architecture includes multiple convolutional layers, with Layer 2 (L2) being the second such layer. L2 replicates the parameters of Layer 1 (L1) but differs in that it does not employ padding, a method identified as 'VALID' padding in many frameworks, including TensorFlow. Both L1 and L2 are subjected to batch normalization to stabilize learning and improve convergence rates. Furthermore, L2 incorporates spatial dropout, a regularization technique that enhances generalization by probabilistically omitting entire feature maps during training, with a dropout rate set to 50%.

After L2, Layer 3 (L3) incorporates a convolutional operation using a reduced kernel size and VALID padding, followed by a one-dimensional average pooling layer (L4). Pooling operations, such as those in L4, are critical in convolutional neural networks (CNNs) for reducing dimensionality, which diminishes computational demands and the model's parameter count. Additionally, such reductions in size confer a degree of translational invariance to the model, facilitating the recognition of patterns regardless of their specific positions within the input space.

Further layers include an additional convolutional layer (L5) with subsequent spatial dropout and a flattening layer (L6), which transforms the matrix format of the data into a vector format. This transformation is pivotal for interfacing with subsequent, non-spatial layers. The architecture concludes with three fully connected layers (L7, L8, and L9), incorporating dropout and consisting of 296, 148, and 74 units, respectively. Each layer ensures dense connectivity to its predecessor, allowing activations $y_j(l)$

$$y_j^{(l)} = f\left(\sum_{i=1}^{I} w_{ji}^{(l)} \cdot x_i^{(l-1)} + b_j^{(l)}\right) \tag{8}$$

where I is the number of units in the previous layer, l is the current layer, $w_{ji}^{(l)}$ is the weight of the connection between unit j of this layer and unit i of the previous layer, $b_j^{(l)}$ is the bias term of unit j, and f is the unit transfer function. With the exception of the output layer, the transfer function f of the units of all layers is a rectified linear unit (ReLU) function:

$$f(x) = \begin{cases} x, & \text{if } x > 0 \\ 0.01 \cdot x, & \text{otherwise} \end{cases} \tag{9}$$

The five units of the last output layer use a softmax function to encode the probabilities $\hat{y}_i$ of the five categories of the MI classification task:

$$\hat{y}_i = \arg\max\left(\frac{e^{y_i}}{\sum_{i=1}^{5} e^{y_i}}\right) \tag{10}$$

The *HopefullNet* model features a sequential convolutional neural network architecture primarily designed for processing 1-dimensional data, as detailed in Table III. The network begins with a convolutional layer (L1) utilizing 32 filters of size 20. This layer uses 'Relu' as the activation function and employs 'SAME' padding, which preserves the input dimensionality by padding the input equally on both sides. Following the convolution, batch normalization (BN) is applied, which normalizes the activations of the previous layer at each batch, stabilizing the learning process by maintaining the mean output close to 0 and the output standard deviation close to 1.

The second layer (L2) mirrors the first in terms of the number of filters and filter size but changes the padding to 'VALID,' which does not pad the input and thus reduces the dimensionality. This layer also includes batch normalization for the reasons mentioned and incorporates spatial dropout. Unlike regular dropout, spatial dropout drops entire 1D feature maps instead of individual elements, which is particularly useful for spatially correlated features in convolutional neural networks.

Continuing to layer L3, the network uses a smaller kernel size of 6 for finer feature extraction, still employing 'Relu' activation and 'VALID' padding. This is immediately followed by an average pooling layer (L4), which helps reduce the data's dimensionality, thus reducing the number of parameters and computations in the network.

Layer L5 applies another convolution with the same settings as L3 and includes spatial dropout to further regularize the model by promoting independent representations in the feature maps.

The flattened layer (L6) transforms the 3D output of previous layers into a 1D array, making it possible to feed it into fully connected layers. The first fully connected layer (L7) has 296 neurons, followed by a dropout layer that helps prevent overfitting by randomly setting neuron activations to zero during training. Subsequent fully connected layers (L8 and L9) reduce in size, further compressing the representation and applying dropout after each layer.

The final layer (L10) is a fully connected layer with 5 neurons corresponding to the model's output classes and uses the softmax activation function. This function converts the logits, the raw prediction values from the last fully connected layer, into probabilities by normalizing them into a probability distribution consisting of potential outcomes for each class.

This architecture efficiently handles various types of 1-dimensional input data, utilizing a combination of convolutional layers for feature extraction and dense layers for classification. The strategic use of dropout and batch normalization throughout the network enhances the model's ability to generalize well to new, unseen data, thus improving its robustness and performance.

F. Experimental Setup

We manually fine-tuned the 1D-CNN, adjusting various parameters to optimize its performance. We experimented with different configurations using the following values as critical aspects.

We opted for categorical cross-entropy loss as our loss function since both hypotheses involve multi-class classification.

Additionally, Adam [cite] stands out as a widely embraced optimization algorithm for neural networks. Given neural networks' tendency to overfit quickly to training data and underperform on unseen data, we implemented batch normalization and dropout layers as regularization mechanisms. Furthermore,

TABLE III
HopefullNet architecture.

Layer	Features	Type	Input size	Filters	Kernel size	Activation
L1	Conv1D, BN	1DConvolution	(640, 2)	32	20	relu
L2	Conv1D, BN, SD	1DConvolution	(640, 32)	32	20	relu
L3	Conv1D, AP	1DConvolution	(621, 32)	32	6	relu
L4	Conv1D, SD	1DConvolution	(308, 32)	32	6	relu
L5	Flatten	Flatten	(303, 32)	-	-	-
L6	Dense, Dropout	Fully-conn.	(9696)	-	-	relu
L7	Dense, Dropout	Fully-conn.	(296)	-	-	relu
L8	Dense, Dropout	Fully-conn.	(148)	-	-	relu
L9	Dense, Dropout	Fully-conn.	(74)	-	-	relu
L10	Dense	Fully-conn.	(74)	5	-	softmax

TABLE IV
Summary of parameters used in the model

Parameter	Used Value
Loss function	Categorical cross-entropy loss function
Optimizer	Adam optimizer
Regularization	Batch normalization and Dropout layers
Learning rate	1×10^{-4}
Training epochs	100
Batch size	10
Early stopping patience	10
Early stopping tolerance	1×10^{-3}

we employed early stopping to counter overfitting and ensure good generalization to unseen data. Lastly, we trained the 1D-CNN using batches of data with a stochastic gradient descent approach.

IV. Results

In this section, we evaluate the performance of our classification model applied to the motor imagery EEG dataset. The model's effectiveness is quantified using several metrics, including Accuracy, Precision, Recall, F1-Score, and the distribution of predictions as represented by the Confusion Matrix. These metrics provide a comprehensive understanding of the model's ability to classify EEG signals across different categories correctly:

Accuracy: Accuracy is the ratio of correctly predicted observations to the total observations.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{10}$$

Precision: Precision is the ratio of correctly predicted positive observations to the total predicted positives.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{11}$$

Recall: Recall is the ratio of correctly predicted positive observations to all observations in the actual class.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{12}$$

Confusion matrix: A confusion matrix summarizes the number of correct and incorrect predictions for each class.

		Predicted	
		Negative	Positive
Actual	Negative	TN	FP
	Positive	FN	TP

In Table IV, we present the classification performance metrics of our proposed model under Hypothesis 1. The table summarizes the precision, recall, F1-score, and support for each class evaluated in the study. Notably, Class B exhibits a relatively lower precision of 0.93 but achieves perfect recall at 1.00, resulting in an F1-score of 0.96 with a support of 5562 instances. This suggests that while Class B has some instances of over-prediction, it successfully identifies all relevant cases. In contrast, Class LI shows a precision of 1.00 and a recall of 0.91, culminating in an F1-score of 0.95 with 1395 instances, indicating very high accuracy in prediction but slightly missing some positive cases.

Classes RI and LR both maintain high precision and recall values, with F1-scores of 0.96, supported by 1394 and 1368 instances, respectively. Class RR matches the performance excellence of Class LI, with a precision of 1.00 and recall of 0.92, reflecting similar predictive accuracy and minor misses in positive class identification.

At a more aggregated level, the macro average metrics across all classes are 0.98 for precision, 0.94 for recall, and 0.96 for the F1-score, suggesting robust overall model performance. The weighted averages, accounting for the support of each class, are consistently 0.96 across all three metrics, indicating a balanced performance across classes with varying numbers of instances.

The overall accuracy of the model stands at an impressive 0.9607, underscoring the effectiveness of the model in classifying instances across different categories under the tested hypothesis. This high level of accuracy, coupled with the detailed performance metrics, confirms the robustness of the model in handling diverse data representations, making it a reliable tool for further deployment in related tasks.

The confusion matrix in Figure 4 shows the classification performance of our 1D-CNN model. The confusion matrix presented here illustrates the classification accuracy of a model across five classes: B, LI, RI, LR, and RR. Each row of the matrix corresponds to the actual class labels, and each column represents the predicted labels. The matrix highlights

TABLE V
CLASSIFICATION REPORT FOR HYPOTHESIS 1

Class	Precision	Recall	F1-score	Support
B	0.93	1.00	0.96	5562
LI	1.00	0.91	0.95	1395
RI	0.99	0.94	0.96	1394
LR	0.99	0.92	0.96	1368
RR	1.00	0.92	0.96	1388
Macro Avg	0.98	0.94	0.96	11107
Weighted Avg	0.96	0.96	0.96	11107
Overall Accuracy: 0.9607				
Loss: 0.1251				

strong diagonal values, indicating a high number of correct predictions for each class, such as 5550 correct predictions for class B and 1274 for class LI. The model shows particularly strong performance in accurately predicting class B and class RR, with very few instances misclassified. However, there is some notable confusion between other classes, such as class LI being misclassified as class B in 117 instances, and class LR and RR each misclassifying around 100 instances as class B. Despite these occasional confusions, the matrix generally shows that the model performs robustly with high true positive rates across the board and minimal errors, suggesting effective class distinction capability within the model.

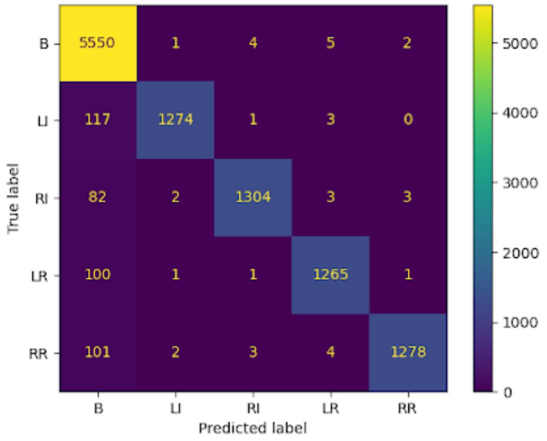

Fig. 4. Classification confusion matrix for the hypothesis 1

In Table V, we detail the classification performance metrics of our model under Hypothesis 2, showcasing precision, recall, F1-score, and support for each class. Class B achieved a precision of 0.96 and a recall of 0.99, resulting in an F1-score of 0.98 with a large support of 5556 instances, indicating highly accurate identification with minimal false positives. Class BI exhibited exceptional precision at 0.99, though the recall was slightly lower at 0.94, leading to an F1-score of 0.96 across 1377 instances, suggesting that while nearly all predictions were correct, some actual positives were missed. Classes FI and BR both demonstrated strong metrics with precision and recall values over 0.95 and F1-scores of 0.96, supported by 1384 and 1397 instances respectively, indicating effective balance in detection and classification. Class FR also showed reliable classification capabilities with a precision of 0.97 and recall of 0.95, yielding an F1-score of 0.96 from 1380 instances.

The aggregated metrics further highlight the model's effectiveness; the macro averages across all classes are 0.97 for precision, 0.95 for recall, and 0.96 for the F1-score, illustrating a high consistency in performance. The weighted averages are consistently 0.97 for precision, recall, and F1-score, taking into account the support for each class, pointing to well-balanced performance across varying data distributions. The overall model accuracy stands at an impressive 0.9689 with a loss of 0.1519, underlining the model's robustness and practical applicability in handling complex classification tasks, confirming its suitability for deployment in scenarios where high precision and reliability are critical.

TABLE VI
CLASSIFICATION REPORT FOR HYPOTHESIS 2

Class	Precision	Recall	F1-score	Support
B	0.96	0.99	0.98	5556
BI	0.99	0.94	0.96	1377
FI	0.98	0.94	0.96	1384
BR	0.97	0.95	0.96	1397
FR	0.97	0.95	0.96	1380
Macro Avg	0.97	0.95	0.96	11094
Weighted Avg	0.97	0.97	0.97	11094
Overall Accuracy: 0.9690				
Loss: 0.1519				

In Figure 5 we present the classification performance of our 1D-CNN model. The confusion matrix displayed visualizes the performance of a classification model across five classes: B, BI, FI, BR, and FR. The diagonal elements represent the number of instances correctly predicted for each class, showing high values and thus indicating strong model accuracy. For instance, class B has 5515 correct predictions, which dominates its row, reflecting high true positive rates. The off-diagonal elements, however, represent misclassifications. While these numbers are significantly lower than the diagonal values, they highlight areas where the model confuses one class for another. For example, class B has 12 instances misclassified as FI, and class BR has 20 instances misclassified as FR. Despite these misclassifications, the overall diagonal dominance suggests that the model performs well, particularly in distinguishing between most classes with only minor confusion between some, like between BR and FR where there are 20 misclassifications.

V. DISCUSSION

The results obtained from testing the hypotheses offer valuable insights into the efficacy of 1D CNNs for distinguishing between real and imaginary motions of various body parts based on EEG signals.

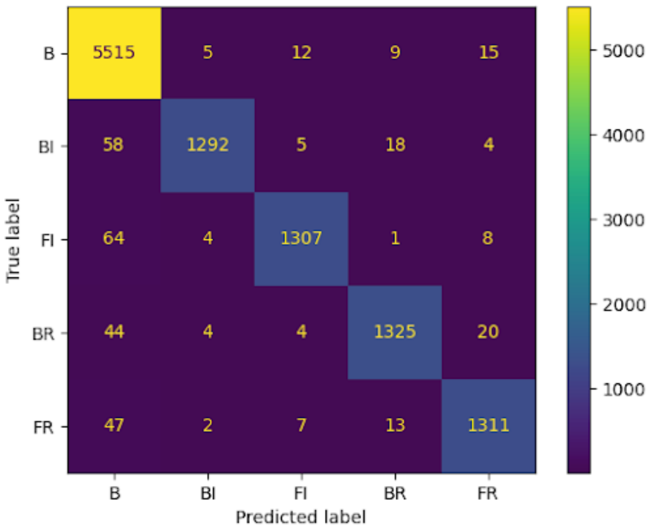

Fig. 5. Classification confusion matrix for the hypothesis 2

For the first hypothesis, which focused on distinguishing between real and imaginary motions of the right and left fists, the achieved accuracy was impressively high at approximately 96.07%. The classification report further elucidates the model's performance across different classes, demonstrating consistently high precision, recall, and F1 scores across the board. Notably, the model exhibited particularly strong performance in classifying the 'B' (Baseline) class, achieving a precision of 93% and full recall. These results indicate that the 1D CNN approach, with minimal preprocessing and data alteration, effectively discriminates between EEG signals corresponding to real and imaginary motions of the fists.

Moving on to the second hypothesis, which expanded the classification task to include motions of both fists and both feet, the accuracy improved slightly to approximately 96.90%. The classification report reveals a similarly robust performance across all classes, with high precision, recall, and F1 scores. Notably, the addition of foot motions did not significantly impact the model's ability to discern between real and imaginary motions, as evidenced by the consistently high-performance metrics. The confusion matrix visually represents the model's classification performance, illustrating the distribution of correct and incorrect classifications across different classes.

Overall, these results support the hypotheses and demonstrate the effectiveness of 1D CNNs for distinguishing between real and imaginary motor imagery tasks based on EEG signals. The high accuracy and robust performance metrics underscore the potential of this approach for various applications in brain-computer interface systems and motor rehabilitation technologies. However, further investigation and validation on larger and more diverse datasets may be warranted to assess the generalizability and scalability of the proposed method fully.

VI. Conclusion

The outcomes of this study robustly support the effectiveness of 1D Convolutional Neural Networks (CNNs) in distinguishing between real and imagined motor movements from EEG signals with minimal data preprocessing. Our results confirmed both hypotheses, demonstrating high classification accuracies and F1-scores. Specifically, the first hypothesis involving right and left fist movements resulted in an accuracy of 96.07% and an F1-score of 0.96, indicating precise model performance with a low error rate. The extension of the model in the second hypothesis to include both fists and feet movements also showed excellent accuracy (96.90%) and an improved F1-score (0.97), albeit with a slightly higher loss. These findings underscore the potential of using simple yet powerful neural network architectures for complex pattern recognition tasks in neural data, paving the way for future advancements in brain-computer interfaces and rehabilitative technologies. Further research could explore the scalability of this approach to other types of EEG-based classifications and its real-world application across different patient groups and movement types.

VII. Future Work

Based on the results obtained and considering the potential for further improvement and exploration, several future directions can be considered:

1) Exploration of Event-Related Potentials (ERPs) with 1D-CNNs: Investigate the incorporation of event-related potentials (ERPs) alongside EEG signals in the classification task using 1D CNNs. ERPs represent brain responses to specific stimuli or tasks and could provide additional discriminative features for improving classification accuracy.
2) Addressing Class Imbalance: Implement data augmentation techniques, such as those similar to the Synthetic Minority Over-sampling Technique (SMOTE) utilized by Mattioli et al., to address class imbalance issues. This approach generates synthetic samples for minority classes to balance the data distribution and improve the model's generalization ability across all classes.
3) Fine-Tuning Hyperparameters: Conduct a thorough hyperparameter tuning process to optimize the 1D CNN model's performance further. Fine-tuning hyperparameters, including learning rate, batch size, kernel size, and number of layers, can help enhance classification results by refining the model's architecture and training process.
4) Simplified Classification Task: Consider simplifying the classification task by focusing solely on distinguishing between the four labels (Right Fist, Left Fist, Right Foot, Left Foot) without including the resting state. By removing the resting state class, the model can focus exclusively on discriminating between different motor imagery tasks, potentially leading to improved classification accuracy and reduced complexity.

By pursuing these future directions, researchers can enhance the effectiveness and robustness of 1D CNN-based classification models for distinguishing between real and imaginary motor imagery tasks based on EEG signals. These efforts could contribute to advancements in brain-computer interface systems, neurorehabilitation technologies, and other applications aimed at decoding neural activity for motor control and communication.

Acknowledgment

We would like to thankfully acknowledge Dr. Katy Tarrit Mirakhorli for her continuous support and valuable guidance throughout this project. The technical support and advanced computing resources from the University of Hawaii Information Technology Services – Cyberinfrastructure, funded in part by the National Science Foundation CC* awards #2201428 and #2232862, are gratefully acknowledged.

References

[1] N. A. Alzahab, L. Apollonio, A. Di Iorio, M. Alshalak, S. Iarlori, F. Ferracuti, A. Monteriù, and C. Porcaro, “Hybrid deep learning (hdl)-based brain-computer interface (bci) systems: a systematic review,” *Brain sciences*, vol. 11, no. 1, p. 75, 2021.

[2] G. Di Pino, C. Porcaro, M. Tombini, G. Assenza, G. Pellegrino, F. Tecchio, and P. M. Rossini, “A neurally-interfaced hand prosthesis tuned inter-hemispheric communication,” *Restorative neurology and neuroscience*, vol. 30, no. 5, pp. 407–418, 2012.

[3] M. Tombini, J. Rigosa, F. Zappasodi, C. Porcaro, L. Citi, J. Carpaneto, P. M. Rossini, and S. Micera, “Combined analysis of cortical (eeg) and nerve stump signals improves robotic hand control,” *Neurorehabilitation and neural repair*, vol. 26, no. 3, pp. 275–281, 2012.

[4] G. A. M. Vasiljevic and L. C. De Miranda, “Brain–computer interface games based on consumer-grade eeg devices: A systematic literature review,” *International Journal of Human–Computer Interaction*, vol. 36, no. 2, pp. 105–142, 2020.

[5] D. Marshall, D. Coyle, S. Wilson, and M. Callaghan, “Games, gameplay, and bci: the state of the art,” *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 5, no. 2, pp. 82–99, 2013.

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

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

[8] R. Biran, D. C. Martin, and P. A. Tresco, “Neuronal cell loss accompanies the brain tissue response to chronically implanted silicon microelectrode arrays,” *Experimental neurology*, vol. 195, no. 1, pp. 115–126, 2005.

[9] R. J. Vetter, J. C. Williams, J. F. Hetke, E. A. Nunamaker, and D. R. Kipke, “Chronic neural recording using silicon-substrate microelectrode arrays implanted in cerebral cortex,” *IEEE transactions on biomedical engineering*, vol. 51, no. 6, pp. 896–904, 2004.

[10] N. Weiskopf, K. Mathiak, S. W. Bock, F. Scharnowski, R. Veit, W. Grodd, R. Goebel, and N. Birbaumer, “Principles of a brain-computer interface (bci) based on real-time functional magnetic resonance imaging (fmri),” *IEEE transactions on biomedical engineering*, vol. 51, no. 6, pp. 966–970, 2004.

[11] K.-S. Hong, N. Naseer, and Y.-H. Kim, “Classification of prefrontal and motor cortex signals for three-class fnirs–bci,” *Neuroscience letters*, vol. 587, pp. 87–92, 2015.

[12] G. Ranky and S. Adamovich, “Analysis of a commercial eeg device for the control of a robot arm,” in *Proceedings of the 2010 IEEE 36th Annual Northeast Bioengineering Conference (NEBEC)*, pp. 1–2, IEEE, 2010.

[13] F. Carrino, J. Dumoulin, E. Mugellini, O. Abou Khaled, and R. Ingold, “A self-paced bci system to control an electric wheelchair: Evaluation of a commercial, low-cost eeg device,” in *2012 ISSNIP biosignals and biorobotics conference: biosignals and robotics for better and safer living (BRC)*, pp. 1–6, IEEE, 2012.

[14] F. Ferracuti, V. Casadei, I. Marcantoni, S. Iarlori, L. Burattini, A. Monteriù, and C. Porcaro, “A functional source separation algorithm to enhance error-related potentials monitoring in noninvasive brain-computer interface,” *Computer Methods and Programs in Biomedicine*, vol. 191, p. 105419, 2020.

[15] F. Ferracuti, A. Freddi, S. Iarlori, S. Longhi, A. Monteriù, and C. Porcaro, “Augmenting robot intelligence via eeg signals to avoid trajectory planning mistakes of a smart wheelchair,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, no. 1, pp. 223–235, 2023.

[16] F.-B. Vialatte, M. Maurice, J. Dauwels, and A. Cichocki, “Steady-state visually evoked potentials: focus on essential paradigms and future perspectives,” *Progress in neurobiology*, vol. 90, no. 4, pp. 418–438, 2010.

[17] A. Combaz and M. M. Van Hulle, “Simultaneous detection of p300 and steady-state visually evoked potentials for hybrid brain-computer interface,” *PLoS One*, vol. 10, no. 3, p. e0121481, 2015.

[18] S. Salelkar and S. Ray, “Interaction between steady-state visually evoked potentials at nearby flicker frequencies,” *Scientific Reports*, vol. 10, no. 1, p. 5344, 2020.

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

[20] G. Schalk, D. J. McFarland, T. Hinterberger, N. Birbaumer, and J. R. Wolpaw, “Bci2000: a general-purpose brain-computer interface (bci) system,” *IEEE Transactions on biomedical engineering*, vol. 51, no. 6, pp. 1034–1043, 2004.

[21] C. Hall, “Individual differences in the mental practice and imagery of motor skill performance,” *Canadian journal of applied sport sciences. Journal canadien des sciences appliquees au sport*, vol. 10, no. 4, pp. 17S–21S, 1985.

[22] F. Lotte, C. Jeunet, J. Mladenović, B. N'Kaoua, and L. Pillette, “A bci challenge for the signal processing community: considering the user in the loop,” *Signal Processing and Machine Learning for Brain-Machine Interfaces*, pp. 1–33, 2018.

[23] J. Ruan, X. Wu, B. Zhou, X. Guo, and Z. Lv, “An automatic channel selection approach for ica-based motor imagery brain computer interface,” *Journal of Medical Systems*, vol. 42, pp. 1–13, 2018.

[24] G. Barbati, C. Porcaro, F. Zappasodi, P. M. Rossini, and F. Tecchio, “Optimization of an independent component analysis approach for artifact identification and removal in magnetoencephalographic signals,” *Clinical Neurophysiology*, vol. 115, no. 5, pp. 1220–1232, 2004.

[25] C. Porcaro, M. T. Medaglia, and A. Krott, “Removing speech artifacts from electroencephalographic recordings during overt picture naming,” *NeuroImage*, vol. 105, pp. 171–180, 2015.

[26] A. Delorme and S. Makeig, “Eeglab: an open source toolbox for analysis of single-trial eeg dynamics including independent component analysis,” *Journal of neuroscience methods*, vol. 134, no. 1, pp. 9–21, 2004.

[27] B. Xu, L. Zhang, A. Song, C. Wu, W. Li, D. Zhang, G. Xu, H. Li, and H. Zeng, “Wavelet transform time-frequency image and convolutional network-based motor imagery eeg classification,” *Ieee Access*, vol. 7, pp. 6084–6093, 2018.

[28] P. Gaur, H. Gupta, A. Chowdhury, K. McCreadie, R. B. Pachori, and H. Wang, “A sliding window common spatial pattern for enhancing motor imagery classification in eeg-bci,” *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–9, 2021.

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

[30] C. Porcaro, G. Coppola, G. Di Lorenzo, F. Zappasodi, A. Siracusano, F. Pierelli, P. M. Rossini, F. Tecchio, and S. Seri, “Hand somatosensory subcortical and cortical sources assessed by functional source separation: an eeg study,” *Human brain mapping*, vol. 30, no. 2, pp. 660–674, 2009.

[31] C. Porcaro and F. Tecchio, “Semi-blind functional source separation algorithm from non-invasive electrophysiology to neuroimaging,” in *Blind Source Separation: Advances in Theory, Algorithms and Applications*, pp. 521–551, Springer, 2014.

[32] Q. Huang, Z. Zhang, T. Yu, S. He, and Y. Li, “An eeg-/eog-based hybrid brain-computer interface: Application on controlling an integrated

wheelchair robotic arm system," *Frontiers in neuroscience*, vol. 13, p. 1243, 2019.

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

[34] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[35] G. Dai, J. Zhou, J. Huang, and N. Wang, "Hs-cnn: a cnn with hybrid convolution scale for eeg motor imagery classification," *Journal of neural engineering*, vol. 17, no. 1, p. 016025, 2020.

[36] B. Sun, X. Zhao, H. Zhang, R. Bai, and T. Li, "Eeg motor imagery classification with sparse spectrotemporal decomposition and deep learning," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 2, pp. 541–551, 2020.

[37] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: concepts, cnn architectures, challenges, applications, future directions," *Journal of big Data*, vol. 8, pp. 1–74, 2021.

[38] S. A. Raurale, G. B. Boylan, S. R. Mathieson, W. P. Marnane, G. Lightbody, and J. M. O'Toole, "Grading hypoxic-ischemic encephalopathy in neonatal eeg with convolutional neural networks and quadratic time–frequency distributions," *Journal of Neural Engineering*, vol. 18, no. 4, p. 046007, 2021.

[39] H. Abbasi, A. J. Gunn, C. P. Unsworth, and L. Bennet, "Advanced deep learning spectroscopy of scalogram infused cnn classifiers for robust identification of post-hypoxic epileptiform eeg spikes," *Advanced Intelligent Systems*, vol. 3, no. 2, p. 2000198, 2021.

[40] R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, and T. Ball, "Deep learning with convolutional neural networks for eeg decoding and visualization," *Human brain mapping*, vol. 38, no. 11, pp. 5391–5420, 2017.

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

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

[43] X. Lun, Z. Yu, T. Chen, F. Wang, and Y. Hou, "A simplified cnn classification method for mi-eeg via the electrode pairs signals," *Frontiers in Human Neuroscience*, vol. 14, p. 338, 2020.

[44] G. Zoumpourlis and I. Patras, "Motor imagery decoding using ensemble curriculum learning and collaborative training," in *2024 12th International Winter Conference on Brain-Computer Interface (BCI)*, pp. 1–8, IEEE, 2024.

[45] K. Roots, Y. Muhammad, and N. Muhammad, "Fusion convolutional neural network for cross-subject eeg motor imagery classification," *Computers*, vol. 9, no. 3, p. 72, 2020.

[46] W. Fadel, C. Kollod, M. Wahdow, Y. Ibrahim, and I. Ulbert, "Multi-class classification of motor imagery eeg signals using image-based deep recurrent convolutional neural network," in *2020 8th International Winter Conference on Brain-Computer Interface (BCI)*, pp. 1–4, IEEE, 2020.

[47] C. M. Köllőd, A. Adolf, K. Iván, G. Márton, and I. Ulbert, "Deep comparisons of neural networks from the eegnet family," *Electronics*, vol. 12, no. 12, p. 2743, 2023.

[48] A. G. Lazcano-Herrera, R. Q. Fuentes-Aguilar, A. Ramirez-Morales, and M. Alfaro-Ponce, "Bilstm and squeezenet with transfer learning for eeg motor imagery classification: Validation with own dataset.," *IEEE Access*, 2023.

[49] F. Mattioli, C. Porcaro, and G. Baldassarre, "A 1d cnn for high accuracy classification and transfer learning in motor imagery eeg-based brain-computer interface," *Journal of Neural Engineering*, vol. 18, no. 6, p. 066053, 2022.

[50] TMSi, "The 10-20 system for eeg." https://info.tmsi.com/blog/the-10-20-system-for-eeg.

[51] "Meg+eeg analysis." https://mne.tools/stable/index.html.