Neural Signal Decoding for Motor Imagery

1st Neelam Karki,
Department of Computer Science
University of South Dakota
Vermillion, South Dakota, USA
neelam.karki@coyotes.usd.edu

2nd Rahisha Pokharel
Department of Computer Science
University of South Dakota
Vermillion, South Dakota, USA
rahisha.pokharel@coyotes.usd.edu

3rd Samita Neupane
Department of Computer Science
University of South Dakota
Vermillion, South Dakota, USA
samita.neupane@coyotes.usd.edu

***Abstract*—Developing efficient Brain-computer Interfaces (BCIs) to read non-linguistic stimuli like motor activities and decipher human intentions from brain activity is still very difficult. In this work, Electroencephalogram (EEG) signals from the PhysioNet Motor Movement/Imagery Dataset (EEGMMIDB) are used to decode motor intents, namely imagined left and right hand movements. EEG signals were preprocessed by bandpass filtering and epoch segmentation, and then deep learning models were used for categorization. Transformer-based models for learning long-range temporal dependencies and Convolutional Neural Networks (CNNs) for capturing spatial data were the two architectures created and compared. To improve classification performance, spatial-spectral-temporal features were also extracted. After 5-fold cross-validation and hyperparameter adjustment, a hybrid CNN-Transformer architecture got the highest test accuracy of 63.27%, although the accuracy of individual CNN and Transformer models was only about 61%. Despite the difficulties caused by noisy, non-stationary EEG signals, and significant inter-subject variability, the results demonstrate the viability of deep learning techniques for motor imagery decoding. In addition to supporting the future development of assistive technologies and neurorehabilitation tools, this work advances the rapidly developing field of brain decoding.**

***Index Terms*—EEG, Motor Imagery, Neural Decoding, PhysioNet, CNN, Transformer**

I. Introduction

The fundamental question that arises in neuroscience is how to read perpetual content stored in the human brain, basically decoding the human brain. In the past days, decoding brain activities using functional magnetic resonance imaging (fMRI) has been a hot topic. But today, the Brain Computer Interface (BCI) System is becoming a burning topic as it provides direct communication between the human brain and external devices, offering significant promise for applications in assistive technologies. This technology holds more importance particularly for individuals with motor impairments who can benefit from hands-free control systems. Among various BCI models, Motor Imagery (MI) has gained substantial attention due to its non-invasive approach and capacity to decode imagined movements from electroencephalographic (EEG) signals. Though it has caused such a speculation, EEG-based decoding is inherently challenging due to the low signal-to-noise ratio, artifacts, and variability between subjects and sessions. Preprocessing, Feature extraction, and Classification play crucial roles in addressing these challenges and improving decoding accuracy. In BCI research, the Common Spatial Pattern (CSP) algorithm has been vastly implemented for feature extraction, as it maximizes the variance differences between classes by transforming the EEG data into a spatial domain. This study leverages the PhysioNet EEG Motor Movement/Imagery Dataset [1], which includes annotated EEG recordings during imagined and executed motor tasks. A preprocessing pipeline was implemented consisting of the steps such as, bandpass filtering to isolate motor-related neural oscillations, epoching of EEG signals time-locked to task cues to capture relevant activity windows, and Common Spatial Pattern (CSP)-based spatial filtering to enhance class-discriminative spatial features. Using the extracted features, the performance of three CNN models, a transformer model, and a hybrid model is compared on a 2-class classification task: left-hand imagery, right-hand imagery. The evaluation includes both test accuracy and loss calculation to assess generalization. This study investigates the potential of Convolutional Neural Networks (CNNs) and Transformer architectures to imrpove classification performance in motor imagery tasks, focusing on differentiating left-hand and right-hand imagery.

A. Objectives

The major objective of this research is to develop a system that can assist in decoding neural signals or brain activities to enable communication of control systems, especially for people with motor disabilities.

Specific Objectives:

- To design and propose an architecture to decode the neural signals associated with motor imagery tasks like hand movements.
- To implement a method for noise filtration in order to enhance the quality of neural signal interpretation.
- To develop and evaluate a deep learning model that combines Convolutional Neural Networks (CNNs) for spatial feature extraction and Transformers for temporal dependency modeling.

II. Related Works

The ability to elicit visible neural patterns in the sensorimotor cortex made motor imagery (MI), the mental practice of physical tasks without overt execution, a key paradigm in brain-computer interface (BCI) research. Recent advancements in signal gathering, processing methods, and clinical applications have significantly enhanced the functionality and performance of MI-based BCIs.

A. Signal Acquisition and Neural Mechanisms

When motor imagery occurred, two cortical areas that overlapped with those involved in real movement were activated: the primary motor cortex and the supplementary motor area. Recent studies have shown that high-resolution fMRI-EEG fusion can decode fine-grained motor imagery tasks, such as individual finger motions, with unprecedented spatial accuracy [2]. Additionally, gamma-band oscillations ($> 60\,\text{Hz}$) were detected in electrocorticography (ECoG) recordings of MI; these oscillations were usually associated with overt movement. According to [3], this implied that the neural fingerprints were more complex than previously believed.

Signal acquisition innovations have also been notable. Since dry EEG electrodes don't need conductive gels, their classification accuracy is now on par with that of conventional wet electrodes, increasing its usefulness for longer-term applications [4]. Mobile BCI devices that used cellphones further enabled real-time MI decoding, expanding accessibility beyond laboratory settings [5].

B. Decoding Algorithms: From Deep Learning to Conventional ML

Earlier MI-BCIs combined manually constructed characteristics (such bandpower and wavelet treatments) with classifiers like Support Vector Machines (SVMs). These methods are useful, but they often struggle with non-stationary EEG data and cross-subject variability. Recent advances in deep learning have revolutionized MI decoding: robust and hierarchical features may be automatically learned from raw EEG signals by deep neural networks, especially Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). The spatiotemporal patterns of motor imagery tasks have been better captured by these models, which has greatly increased classification accuracy and subject-to-subject generalization. Further pushing the limits of MI-BCI performance are designs like deep CNNs, hybrid CNN-RNN models, and attention-based Transformers, which provide a more flexible and scalable solution than conventional techniques. EEG-Transformers used self-attention mechanisms to model long-range interdependence in EEG signals, achieving state-of-the-art accuracy on benchmark datasets [6]. Lightweight CNNs with less than 50k parameters made it possible to construct real-time applications with low latency on edge devices like Raspberry Pi [5]. Hybrid CNN-LSTM architectures improved subject-to-subject generalization by integrating temporal and spatial feature learning [7]. Generative Adversarial Networks (GANs) were utilized to improve model training for small datasets by synthesizing realistic EEG data, which helped alleviate data scarcity [8]. To further enhance decoding performance, Graph Signal Processing (GSP) frameworks were employed to describe EEG channels as networks, optimizing spatial filtering [9].

C. Clinical Uses and Applications

The validation of MI-BCIs in clinical settings is growing. ERD/ERS patterns are correlated with recovery outcomes, and systems that combine MI with real-time feedback have demonstrated encouraging results in restoring motor function for stroke rehabilitation [10]. With greater than 85% accuracy in real-world situations, hybrid EEG-EMG systems also give amputees intuitive prosthetic control [11].

III. Dataset Exploration

The initial phase of the research involved analyzing multiple publicly available EEG datasets to identify the one most aligned with the objectives of motor imagery classification. The EEG Motor Movement/Imagery Dataset from PhysioNet is the one that was finally selected since it contains high-quality, well-annotated EEG recordings that work with well-known signal processing programs like MNE-Python. This dataset offers an enormous amount of information for researching brain activity during imaging and motor tasks. With data from 109 participants, it guarantees a wide variety of brain signal patterns, which is essential for extrapolating results in domains such as brain-computer interfaces (BCIs). Both real motor movements (left and right hand and foot) and motor imagery tasks (mental visualization of movements) are included in the dataset. These distinctions are especially useful for creating BCIs that rely on actions that are thought rather than performed. Comprising of 19,670 usable samples in total, the dataset records fine-grained spatial features of cerebral activity and was recorded using a 64-channel EEG setup that adheres to the 10-10 international electrode placement scheme. While the EDF+ file format guarantees interoperability with a range of analytic tools, the 160 Hz sampling rate strikes a compromise between temporal precision and reasonable data volumes. Every trial follows a set protocol, which starts with a baseline rest period during which the participant does not move, creating a reference state for tasks that follow. The cue phase comes next, during which a visual or aural cue marks the beginning of an imaging or motor execution job. The subject executes or visualizes the movement for four seconds during the task phase. In order to allow brain activity to return to baseline before the following trial starts, the trial ends with a post-task rest interval.

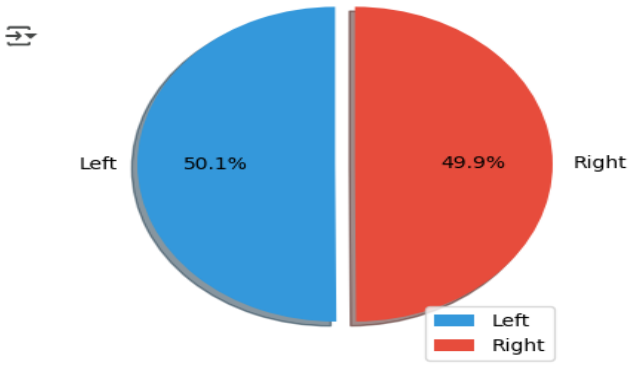

Fig. 1. Class distribution of the two-class motor imagery classification task using the PhysioNet EEG dataset. The plot shows the number of trials corresponding to left-hand (9854 trials) and right-hand (9816 trials) motor imagery, selected from 109 subjects.

IV. System Architecture and Methodology

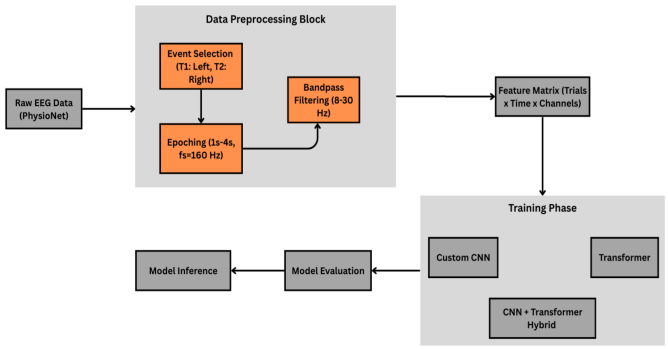

Fig. 2. System architecture for EEG motor imagery classification. The pipeline begins with raw EEG data acquisition from the PhysioNet dataset, followed by preprocessing using bandpass filtering and epoch extraction. Processed signals are then fed into deep learning models (CNN, Transformer, or Hybrid) for classification of motor imagery tasks.

This system implements a structured workflow for motor imagery classification. In this pipeline, raw EEG signals are first preprocessed using bandpass filtering and epoching to isolate brain activity. The processed data are then passed through three different architectures: CNN, Transformer, and a hybrid architecture of CNN and Transformer, where the CNN module extracts spatial features from selected EEG channels, and the transformer module captures temporal dependencies across time steps. Finally, a fully connected classifier interprets the combined features to predict the motor imagery class (left or right hand). Each step of the pipeline is tailored to address the difficulties presented by EEG data, including noise, nonstationarity, and weak signals [5] [6].

A. Data Preprocessing

Preprocessing was performed using the MNE-Python library. Raw EEG signals were loaded into MNE-Python from multiple EDF files that had event annotations and continuous recordings (T1 and T2, which corresponded to left and right fist movements). These signals were bandpass filtered to isolate relevant frequency bands associated with motor imagery (typically within 8-30 Hz). Data were divided into 3-second epochs (1.0-4.0 s after event) without baseline correction after these annotations were transformed into numerical IDs. To reduce dimensionality while retaining critical information, three key EEG channels were selected by prior neurophysiological knowledge. The final representation for each trial was standardized to a shape of [Batch, 1, 3, 641], where 3 represents the selected channels and 641 corresponds to time points over a 4-second window.

B. Data Partitioning and Training Configuration

The dataset was partitioned into training (70%), validation (15%), and testing (15%) subsets to ensure unbiased model evaluation. A batch size of 64 was utilized during training, while 128 was used during validation and testing to optimize GPU memory usage. All models were trained for up to 50 epochs with early stopping applied based on validation loss, using a patience of 5 epochs to prevent overfitting. Training was conducted on Google Colab using an NVIDIA Tesla T4 GPU, which provided adequate computational power for model experimentation and cross-validation.

C. Model Architectures

Three primary deep learning architectures were designed and evaluated for EEG-based motor imagery classification.

1) Convolutional Neural Networks (CNNs): Multiple CNN variants were developed using 2D convolutional layers to extract spatial features from EEG signals. Variations in initial filter sizes, kernel dimensions, and activation functions were tested to optimize performance. All CNN architectures incorporated dropout (ranging from 0.3 to 0.6) and batch normalization to enhance generalization and training stability.

2) Transformer-based Model: A Tranformer architecture was implemented to model the temporal dependencies inherent in EEG time-series data. Input sequences were encoded with sinusoidal positional embeddings and passed through multi-head self-attention layers with GELU activation functions. Layer normalization and global average pooling were included to stabilize learning and reduce dimensionality, enabling the model to capture long-range temporal relationships in the signals.

3) Hybrid CNN + Transformer Model: The hybrid architecture combined CNN and Transformer components to leverage both spatial and temporal information. The CNN module first extracted spatial representations from EEG inputs, which were subsequently processed by Transformer encoder layers to capture temporal dynamics. The model employed dropout layers and positional encoding throughout the pipeline for regularization and effective sequence modeling. This design capitalized on the spatial inductive bias of convolutional filters and the global context modeling capabilities of self-attention mechanisms.

D. Training Setup and Evaluation Metrics

To train the models efficiently and avoid suboptimal convergence, Adam optimizer was leveraged, which adapts learning rates for each parameter based on the first and second moment estimates of gradients. The parameter update is defined as:

$$\theta_t = \theta_{t-1} - \eta \cdot \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} \tag{1}$$

where:

- θ_t is the parameter at time t,
- η is the learning rate,
- $\hat{m}_t$ and $\hat{v}_t$ are the bias-corrected estimates of the first and second moments,
- ϵ is a small constant (typically 10^{-8}) for numerical stability.

An initial learning rate of 0.001 and weight decay of 0.01 for L2 regularization were used. No learning rate scheduler

was applied in the base training, but a random hyperparameter search explored variations in learning rate and weight decay.

The loss function used was the categorical cross-entropy loss, suitable for multi-class classification problems, defined as:

$$\mathcal{L}_{CE} = -\sum_{i=1}^{C} y_i \log(\hat{y}_i) \tag{2}$$

where:

- C is the number of classes,
- y_i is the true label (one-hot encoded),
- $\hat{y}_i$ is the predicted probability for class i.

For some experiments, label smoothing was applied with a smoothing factor $\epsilon = 0.1$, which modifies the ground truth distribution to reduce overconfidence and improve generalization. The label-smoothed target is computed as:

$$y_i^{\text{smooth}} = (1 - \epsilon) \cdot y_i + \frac{\epsilon}{C} \tag{3}$$

where:

- y_i is the true label,
- ϵ is the smoothing factor,
- C is the number of classes.

Regarding activation functions, several were experimented with:

- ELU (Exponential Linear Unit):

$$\text{ELU}(x) = \begin{cases} x & \text{if } x > 0 \\ \alpha(e^x - 1) & \text{if } x \leq 0 \end{cases} \tag{4}$$

  used in CNN layers to promote smooth gradients and handle negative activations.

- SiLU (Sigmoid-weighted Linear Unit):

$$\text{SiLU}(x) = x \cdot \sigma(x), \quad \sigma(x) = \frac{1}{1 + e^{-x}} \tag{5}$$

  used in deeper CNN blocks for its dynamic gating behavior.

- GELU (Gaussian Error Linear Unit):

$$\text{GELU}(x) = x \cdot \Phi(x) = x \cdot \frac{1}{2}\left[1 + \text{erf}\left(\frac{x}{\sqrt{2}}\right)\right] \tag{6}$$

used in the Transformer encoder layers for its smooth probabilistic non-linearity.

All convolutional layers were initialized using Kaiming He initialization, while linear layers in fully connected and Transformer blocks used Xavier (Glorot) initialization to maintain variance and promote stable gradients during training.

Dropout was used as a regularization mechanism, with rates ranging from 0.2 to 0.6 depending on the model depth and layer type. Batch Normalization and Layer Normalization were also employed to stabilize training and accelerate convergence.

V. Experiments and Results

A series of experiments were conducted to evaluate the effectiveness of each model, using classification accuracy and the F1 score as evaluation metrics. All experiments used the same data split and training parameters for fairness.

A. CNN Model Variants

1) Experiment 1: Deep CNN with Layer Normalization (CNN1): The first CNN configuration employed a deep convolutional network with 32 initial filters and a kernel size of 14. This model included multiple convolutional blocks with Layer Normalization and SiLU activations. Despite its depth and normalization layers, the model achieved a test accuracy of only 61.50%, suggesting that the complexity did not significantly enhance performance and may have contributed to overfitting, particularly due to the limited number of trials per subject.

2) Experiment 2: Simplified CNN Architecture (CNN2): To reduce complexity, this architecture utilized only 16 initial filters with a larger kernel size of 31 and ELU activation functions. With the objective to capture broader temporal features with fewer parameters, this configuration showed a slight improvement over Experiment 1, yielding a test accuracy of 61.67%. The results suggest that simplifying the model while adjusting the receptive field can help reduce overfitting and improve generalization.

3) Experiment 3: Optimized CNN with Tuned Input Dimensions (CNN3): The architecture was further optimized by fine-tuning the input dimensions and convolutional parameters. This variant retained a relatively shallow design but used optimal kernel sizes and dropout values for regularization. It achieved the highest accuracy among the CNN-only models at 62.28%, indicating that tuning architectural hyperparameters can lead to moderate yet meaningful improvements in motor imagery classification performance.

B. Transformer Model

A Transformer-based architecture was implemented to explore its ability to capture temporal dependencies in EEG signals. The input sequences were downsampled to 160 time steps and encoded with sinusoidal positional encodings. The model consisted of three Transformer encoder layers, each with four self-attention heads and feedforward layers with GELU activation. Despite its architectural strength, the Transformer achieved a test accuracy of 60.76%, which was lower than all CNN configurations.

C. Hybrid CNN-Transformer Model

To combine the benefits of spatial and temporal modeling, a hybrid architecture was designed where a CNN feature extractor was followed by a Transformer encoder. The CNN extracted local spatial features across EEG channels and time, which were then projected and fed into the Transformer to capture long-range temporal dependencies. This architecture initially achieved a test accuracy of 62.76%, outperforming both the standalone CNN and Transformer models.

To validate the robustness of the hybrid model, a 5-fold cross validation experiment was conducted. Each fold included a different random split of the data, ensuring the model's ability to generalize across samples. Table I summarizes the results of the experiment.

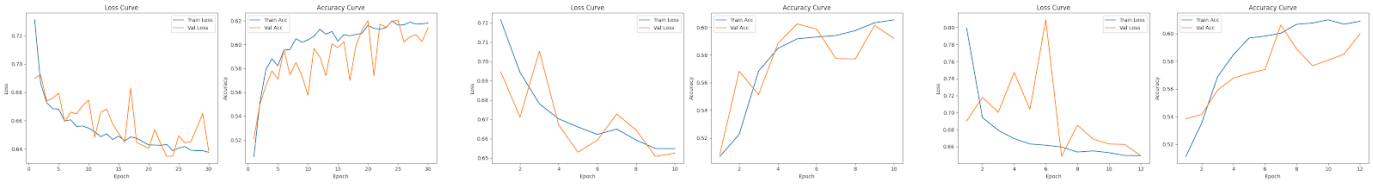

Fig. 3. Training and validation loss and accuracy curves for the three CNN experiments. From left to right: (a) CNN1, (b) CNN2, and (c) CNN3. Each plot illustrates model convergence and generalization across epochs.

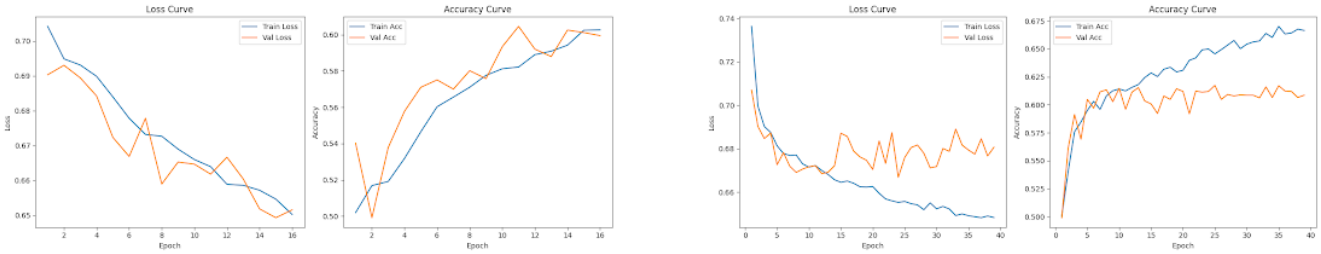

Fig. 4. Training and validation loss and accuracy curves for the Transformer-only model (left) and the CNN–Transformer hybrid model (right). These results emphasize how self-attention mechanisms and hybrid architectures influence model learning dynamics and performance stability.

Hyperparameter tuning was employed to improve the model performance and generalization, where random search was utilized due to its efficiency in high-dimensional search spaces.

The following hyperparameters were explored:

- Learning rate: [1e-4, 2e-4, 3e-4, 1e-3]
- Weight decay: [1e-4, 1e-3]
- Label smoothing: [0.0, 0.1] or [0.0, 0.1]
- Maximum gradient norm: [0.5, 1.0]

TABLE I
CROSS-VALIDATION ACCURACY AND F1 SCORE PER FOLD

Fold	Accuracy	F1 Score
1	61.98%	61.20%
2	62.25%	62.16%
3	63.23%	63.21%
4	63.06%	62.68%
5	60.45%	60.36%
Mean	**62.20% ± 0.0099**	**61.93% ± 0.0103**

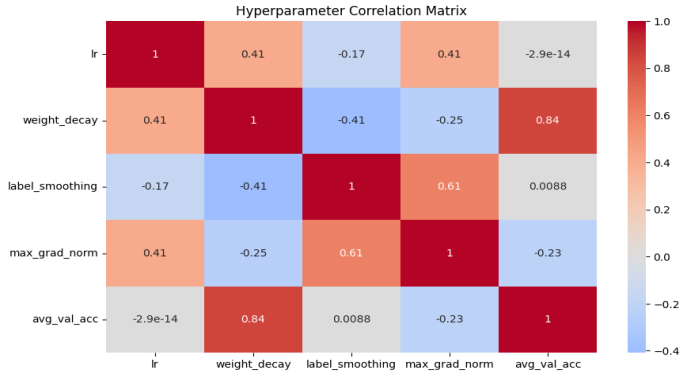

Fig. 5. Correlation heatmap showing the relationship between different parameters and model performance metrics. Warmer colors indicate stronger positive correlations, while cooler colors represent negative correlations.

For each configuration, the model was trained, cross-validated, and evaluated on the same data split. The best-performing setup was:

- Learning rate = 1×10^{-3}
- Weight decay = 0.001
- Label smoothing = 0.1
- Max grad norm = 1.0

Table II presents a summary of the performance metrics across all model variants, including CNN baselines, Transformer, and the hybrid architecture. The hybrid CNN-Transformer model, particularly after hyperparameter tuning, demonstrated the highest test accuracy and most stable cross-validated performance. The improvement over standalone CNN or Transformer models highlights the effectiveness of integrating both spatial and temporal representations for EEG decoding.

TABLE II
COMPARISON OF TEST ACCURACY FOR DIFFERENT MODELS

Model	Test Accuracy (%)
CNN1	61.50
CNN2	61.67
CNN3	62.28
Transformer	60.76
Hybrid (CNN + Transformer)	62.76
Tuned Hybrid	**63.27**

VI. CONCLUSION

This study investigated deep learning approaches for classifying motor imagery tasks from EEG signals using convolutional, attention-based, and hybrid architectures. While CNN models performed reasonably well in capturing spatial information across EEG channels, their performance plateaued

without temporal modeling. The standalone Transformer, although theoretically well-suited for time-series data, underperformed, likely due to its high capacity and lack of spatial inductive biases. The hybrid model emerged as the most effective architecture by integrating CNN-based spatial encoding with Transformer-based temporal processing. After hyperparameter tuning and cross-validation, the hybrid model achieved a peak accuracy of 63.27% outperforming all baseline models. Though the accuracy is modest, it is justified by the noisy, non-stationary, and subject-variable nature of EEG signals. The research emphasizes the value of architectural fusion in deep learning pipelines for EEG decoding and highlight the future potential for personalized and real-time brain-computer interface systems.

VII. Future Work

Although this study achieved promising results with a hybrid deep learning architecture, several research directions remain to be explored. First, subject-specific fine-tuning and transfer learning techniques could be implemented to tailor models to individual EEG patterns, potentially enhancing performance by accounting for inter-subject variability. This would be particularly beneficial in clinical or assistive settings where user-specific adaptation is critical. Additionally, converting one-dimensional EEG signals into two-dimensional spectrograms or scalograms may open up possibilities for using pretrained image-based models such as ResNet or Vision Transformers (ViTs), which have shown strong performance in related domains [12] [13]. Instead of being limited to binary classification, the system's application would be improved by expanding it to multi-class contexts. Finally, testing these models in real-time settings is crucial to bridging the gap between research and practical application.

Acknowledgment

We extend our sincere gratitude to our project supervisor, Dr. Lina Chato, for providing us much needed guidance during this project. We are also deeply thankful to the widely growing machine learning community for their open source contributions, educational resources, which have been indispensable to our learning and experimentation. Finally, we acknowledge everyone who contributed—directly or indirectly—to the success of this project. Your support, whether through discussions, technical assistance, or motivation, is greatly appreciated.

References

[1] A. L. Goldberger, L. A. N. Amaral, L. Glass, *et al.*, "Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, e215–e220, 2000.

[2] K. Bhaganagarapu, G. Jackson, and D. Singh, "Fmri-eeg fusion for finger-specific motor imagery decoding," *Journal of Neuroscience Methods*, vol. 392, p. 109 876, 2023.

[3] K. J. Miller, D. Hermes, and R. P. N. Rao, "Gamma-band oscillations in motor imagery: Ecog evidence," *Nature Neuroengineering*, vol. 5, no. 1, pp. 1–12, 2022.

[4] L. Wang, H. Zhou, and M. Tanaka, "Dry eeg electrodes for high-accuracy motor imagery classification," *Sensors*, vol. 24, no. 3, p. 892, 2024.

[5] Y. Zhang, Q. Li, and W. Chen, "Lightweight cnn for edge-device motor imagery bcis," *IEEE Journal of Biomedical and Health Informatics*, 2024, Advance online publication.

[6] X. Tang, Y. Wang, and X. Gao, "Eeg-transformer: Self-attention for motor imagery decoding," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 31, pp. 2201–2212, 2023.

[7] T. Luo, W. Zhou, and Y. Li, "Cnn-lstm with adaptive temporal kernels for cross-subject mi decoding," *Journal of Neural Engineering*, vol. 19, no. 4, p. 045 003, 2022.

[8] M. Lee, J. Kim, and H. Park, "Wasserstein gans for cross-subject motor imagery decoding," *IEEE Transactions on Biomedical Engineering*, vol. 68, no. 12, pp. 3565–3574, 2021.

[9] R. Zhang, L. Zong, and L. Li, "Graph signal processing for eeg-based motor imagery classification," *Neural Networks*, vol. 156, pp. 51–64, 2022.

[10] S. Bhattacharyya, A. Konar, and D. N. Tibarewala, "Motor imagery bci with real-time feedback for stroke rehabilitation," *Frontiers in Neuroscience*, vol. 16, p. 805 342, 2022.

[11] H. Chen, K. Zhang, and J. Yang, "Hybrid eeg-emg decoding for prosthetic control in amputees," *Scientific Reports*, vol. 13, p. 45 658, 2023.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[13] A. Dosovitskiy, L. Beyer, A. Kolesnikov, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *International Conference on Learning Representations (ICLR)*, 2021.