

EDPNet: An Efficient Dual Prototype Network for Motor Imagery EEG Decoding

Can Han^a, Chen Liu^a, Crystal Cai^a, Jun Wang^{b,*}, Dahong Qian^{a,*}

^a*School of Biomedical Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China*

^b*School of Computer and Computing Science, Hang Zhou City University, Hangzhou, 310015, China*

Abstract

Motor imagery electroencephalograph (MI-EEG) decoding plays a crucial role in developing motor imagery brain-computer interfaces (MI-BCIs). However, decoding intentions from MI remains challenging due to the inherent complexity of EEG signals relative to the small-sample size. In this paper, we propose an Efficient Dual Prototype Network (EDPNet) to enable accurate and fast MI decoding. EDPNet employs a lightweight adaptive spatial-spectral fusion module, which promotes more efficient information fusion between multiple EEG electrodes. Subsequently, a parameter-free multi-scale variance pooling module extracts more comprehensive temporal features. Furthermore, we introduce dual prototypical learning to optimize the feature space distribution and training process, thereby improving the model's generalization ability on small-sample MI datasets. Our experimental results show that the EDPNet outperforms state-of-the-art models with superior classification accuracy and kappa values (84.11% and 0.7881 for dataset BCI competition IV 2a, 86.65% and 0.7330 for dataset BCI competition IV 2b). Additionally, we use the BCI competition III IVa dataset with fewer training data to further validate the generalization ability of the proposed EDPNet. We also achieve superior performance with 82.03% classification accuracy. Benefiting from the lightweight parameters and superior decoding accuracy, our EDPNet shows great potential for MI-BCI applications. The code is publicly available at <https://github.com/hancan16/EDPNet>.

Keywords: prototype learning, attention mechanism, lightweight, brain-computer interface, motor imagery

1. Introduction

Brain-computer interface (BCI) systems enable non-muscular communication between users and machines by interpreting users' neural activity patterns [1]. In BCI applications, Electroencephalography (EEG) has become increasingly popular due to its non-invasive nature and cost-effectiveness. Motor Imagery (MI) [2] is the mental rehearsal of movement execution without any physical movement. When participants visualize moving parts of their body, specific areas of the brain experience energy changes known as event-related desynchronization/synchronization (ERD/ERS). These changes can be recorded via EEG and used to discriminate motor intent [3, 4]. The MI-based BCI has garnered significant attention as it enables the decoding of user motor intentions from EEG signals. It has been successfully applied in various fields, such as stroke rehabilitation [5], wheelchair control [6], and exoskeleton robot arm control [7].

Advancements in deep learning (DL) have significantly increased the accuracy of decoding EEG signals for MI-based BCI applications [8, 9, 10, 11], yet several issues still hinder EEG-based models from reaching practical use [12]. When developing a practical and accurate EEG-MI decoding algorithm, several key challenges need to be taken into consideration.

- **Complex characteristics of EEG.** EEG signals are contaminated with noises and artifacts, leading to a low signal-to-noise ratio (SNR). Moreover, complex spatial-spectral coupling characteristics and high temporal vari-

*Corresponding author.

Email addresses: hancan@sjtu.edu.cn (Can Han), wangjun@hzcu.edu.cn (Jun Wang), dahong.qian@sjtu.edu.cn (Dahong Qian)

ability further complicate the decoding of MI-EEG signals [13]. Therefore, extracting discriminative features from EEG signals is challenging yet crucial.

- **Limited data.** EEG signals frequently encounter constraints due to a scarcity of training samples, caused by several issues such as cumbersome calibration procedures, uncertainty in annotations due to variability in participants' responses to MI tasks, and data privacy issues [14, 15]. Without a massive amount of training data, the model may overfit. This poses challenges to the model's generalization ability on new test data.
- **Computational cost.** In practical BCI applications, computational resources are often limited. Therefore, lightweight and fast models are more suitable for practical scenarios[16].

Existing research primarily focuses on addressing one of the aforementioned challenges. Some research efforts leverage advanced DL techniques, such as multi-branch designs [17, 18, 19], transformers [20, 21, 22, 23, 24], and attention mechanisms [16, 25, 26, 27], to extract highly discriminative features from EEG data, thereby improving EEG-MI decoding accuracy. However, these methods overlook the overfitting issue caused by limited training data and tend to have high computational complexity. Other research efforts [28, 29, 30, 31] employ transfer learning (TL) to mitigate the small-sample issue. Nevertheless, these TL methods still require a relatively large amount of additional data from other subjects to achieve good performance, which may not be practical in real-world scenarios. The Sinc-ShallowNet proposed by Borra et al. [32] offers advantages in lightweight design and interpretability, but its decoding accuracy is unsatisfactory due to the lack of effective mechanisms for extracting discriminative features. Because of incomplete consideration and resolution of these challenges, there remains a gap between existing research and practical applications.

Considering all the above challenges, this paper aims to design a lightweight neural network architecture that effectively extracts highly discriminative and robust features from complex EEG signals for accurate MI classification, even with limited training data. To achieve this goal, we propose an Efficient Dual Prototype Network (EDPNet), inspired by the recognition mechanism of the human brain. Human brains can establish cognitive understanding from a small amount of learnable data and effectively generalize it to new data based on memories and template/prototype matching. The EDPNet is composed of two main components, i.e., a feature extractor and the prototypes for all MI classes. The feature extractor simulates the sensory system of humans for transforming original data into abstract representations. Moreover, the prototypes for each class act as abstract memories of the corresponding class in our brains. As in human recognition, the decision in EDPNet is made by matching the feature (abstract representation) with prototypes (memories) of each class.

For the feature extractor component of EDPNet, based on ERD/ERS prior knowledge, we design two novel modules: Adaptive Spatial-Spectral Fusion (ASSF) module and Multi-scale Variance Pooling (MVP). The ASSF module focuses on modeling the relationship between EEG electrode channels that reflect levels of brain activation [33]. Equipped with a lightweight attention mechanism, the ASSF module can adaptively adjust the weights of each EEG channel according to its importance, thereby effectively extracting spatial-spectral features relevant to specific MI tasks. Then, the MVP module captures multi-scale long-term temporal features based on signal variance which represents the EEG spectral power [34]. The MVP module has no trainable parameters and is computationally efficient, serving as a superior method for extracting powerful temporal features in MI-EEG decoding tasks. The combination of the ASSF module and MVP module enables the feature extractor of EDPNet to extract discriminative spatial-spectral-temporal features from the complex EEG signals.

Furthermore, to address the small-sample dilemma in EEG-MI decoding, we design a new prototype learning (PL) approach to optimize the distribution of prototypes and features, aiming to obtain a robust feature space. To the best of our knowledge, this is the first study to apply the PL to MI-EEG decoding. The classic PL method [35] employs a prototype loss to push feature vectors towards corresponding prototypes, making the features within the same class more compact, which is beneficial for classification and model generalization. Based on the classic PL method, we propose a Dual Prototype Learning (DPL) approach for our EDPNet to decouple inter-class separation and intra-class compactness in training processes. The DPL not only enhances intra-class compactness, but also explicitly increases inter-class margins. Compared to the classic PL, our DPL further improves the model's generalization capability.

The major contributions of this paper can be summarized as follows:

1. Inspired by clinical prior knowledge of EEG-MI and human brain recognition mechanisms, we propose a high-performance, lightweight, and interpretable MI-EEG decoding model EDPNet. The EDPNet simultaneously

considers and overcomes three major challenges in MI-BCIs.

2. To extract highly discriminative features from EEG signals, we design two novel modules, ASSF and MVP, for the feature extractor of EDPNet. The ASSF module extracts effective spatial-spectral features, and the MVP module extracts powerful multi-scale temporal features.
3. To overcome the small-sample issue of MI tasks, we propose a novel DPL approach to optimize the distribution of features and prototypes, aiming to obtain a robust feature space. This enhances the generalization capability and classification performance of our EDPNet.
4. We conduct experiments on three benchmark public datasets to evaluate the superiority of the proposed EDPNet against state-of-the-art (SOTA) MI decoding methods. Additionally, comprehensive ablation experiments and visual analysis demonstrate the effectiveness and interpretability of each module in the proposed EDPNet.

2. Related Works

2.1. Deep Learning based EEG-MI Decoding

With recent advancements in deep learning, researchers are increasingly using various deep learning architectures to decode EEG signals. DeepConvNet [8] employed multiple convolutional layers with temporal and spatial feature extraction kernels. Sakhavi et al. [9] utilized FBCSP for feature extraction, followed by CNN-based classification. Lawhern et al. [10] introduced a compact network, EEGNet, employing depthwise and separable convolution for spatial-temporal feature extraction. However, due to the lack of effective mechanisms for extracting highly discriminative features, the improvements of these methods are limited.

Attention mechanisms, which have recently gained significant recognition in various fields, have been successfully applied to MI-EEG decoding. TS-SEFFNet [25] combines a channel attention module based on the wavelet packet sub-band energy ratio with a temporal attention mechanism, followed by a feature fusion architecture. LMDA-Net [16] combines a custom channel recalibration module with a feature channel attention module from ECA-Net [36]. Wimpff et al. [26] applied various attention mechanisms to the proposed BaseNet and made a very comprehensive comparison of the different variations. M-FANet [27] uses a convolution with a small kernel size to extract local spatial information and a SE [37] module to extract information from multiple perspectives. These methods mainly apply attention mechanisms to deep features extracted by the neural network and improve the MI decoding accuracy to some extent. Nonetheless, few studies have employed attention mechanisms to model the relationships between EEG electrode channels that reflect levels of brain activation [33].

Besides, research efforts have also been devoted to extracting more effective temporal features from high temporal resolution EEG signals. Lately, transformer models have made waves in natural language and computer vision due to the inherent perception of global dependencies [38]. Transformers also emerged in MI decoding and achieved good performance [20, 21, 22, 23, 24], by leveraging long-term temporal relationships. ATCNet [22] uses self-attention to highlight the most important information in EEG signals. Conformer [23] stacks transformer blocks to extract long-term dependency features based on local temporal features extracted by CNN. However, transformer models have high parameters and computational costs, making them hard to be used for real-time MI decoding.

2.2. Prototype Learning

PL simulates the way humans learn by memorizing typical examples to understand and generalize to new situations. In PL methods, a set of representative samples (prototypes) is learned during training, and during testing test samples are assigned to the closest prototype to determine their categories. In [39], Snell et al. proposed to apply the prototype concept for few-shot learning. However, this method learns the prototypes and the feature extractor separately using discriminative loss. Yang et al. [35] introduced the prototype model into the DL paradigm and designed different discriminative loss as well as generative loss. This significantly improves the performance and robustness of DL models in classification tasks. Following the study [35], a substantial amount of research [40, 41, 42] has been devoted to using PL to learn a compact feature space for addressing open-world recognition. Another line of research [43, 44] continues to explore the potential of PL in few-shot learning.

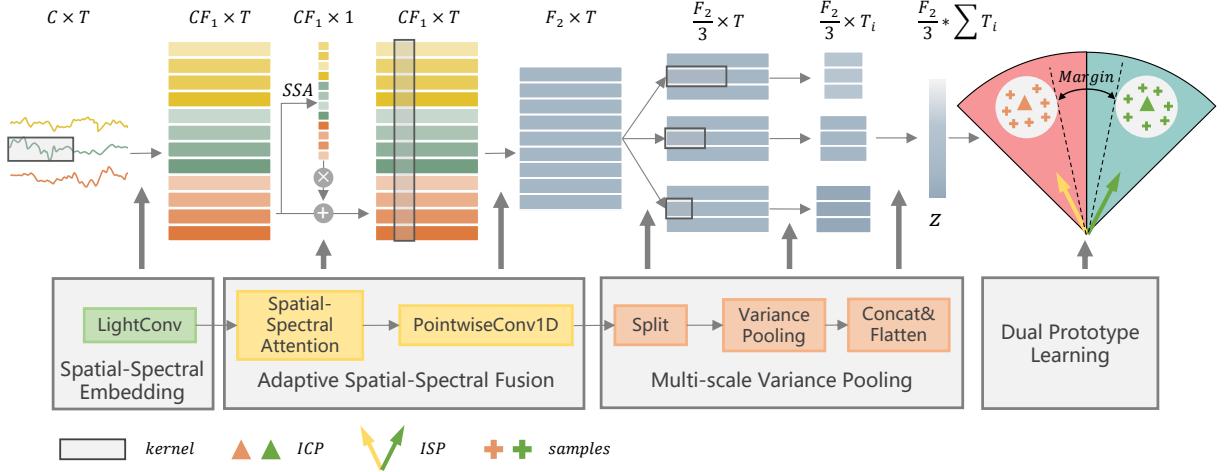


Figure 1: The overall framework of the proposed EDPNet. C and T denote the number of EEG channels and the number of time points, respectively. F_1 and F_2 denote the numbers of temporal filters and spatial-spectral filters, respectively. T_i represents the output length of the variance layer with different kernel sizes. The SSE, ASSF, and MVP make up the feature extractor, while DPL is used for training optimization and classification.

3. Method

As shown in Figure 1, our proposed EDPNet consists of four modules. The Spatial-Spectral Embedding (SSE) module, the Adaptive Spatial-Spectral Fusion module, and the Multi-scale Variance Pooling module constitute the feature extractor for extracting highly discriminative features. The Dual Prototype Learning module is used to optimize the feature space and make classification decisions.

3.1. EEG Representation

In this paper, we feed raw MI-EEG signals into the proposed model without any additional time-consuming preprocessing. Given a set of m labeled MI trials $S = \{X_i, y_i\}_{i=1}^m$, where $X_i \in \mathbb{R}^{C \times T}$ consists of C channels (EEG electrodes) and T time points, $y_i \in \{1, \dots, n\}$ is the corresponding class label, and n is the total number of predefined classes for set S , our EDPNet model first maps a motor imagery trial X_i to the feature space Z and obtains $z_i = f(X_i) \in \mathbb{R}^d$, where f is the feature extractor, as shown in Figure 1. Then, the DPL module maps the feature z_i to its corresponding class y_i .

3.2. Feature Extractor of EDPNet

3.2.1. Spatial-Spectral Embedding

Since different MI classes may differ in their corresponding spectral-spatial sensorimotor rhythm (SMR) patterns [10], most existing studies first extract multi-view spectral information from each EEG electrode channel to form a spatial-spectral representation. Some works follow the practice of EEGNet [10] and use a 2D convolution to extract spectral features, while others follow the practice of FBCNet [9] and use multiple narrow-band digital filters to manually extract different spectral features. Our SSE module is similar to the former, but uses a 1D convolution for end-to-end extraction of spectral features.

Unlike these popular methods [16, 22, 23, 25, 26, 27] following EEGNet, we do not incorporate an additional feature dimension to create a 2D representation $X_i \in \mathbb{R}^{1 \times C \times T}$ for the raw EEG signal $X_i \in \mathbb{R}^{C \times T}$. We directly treat the EEG electrode dimension C in $X_i \in \mathbb{R}^{C \times T}$ as the feature dimension and use 1D convolution to extract spectral features. Specifically, we introduce the LightConv [45], a depthwise convolution that shares certain output channels, to act as the 1D temporal convolution. LightConv first divides the input signal $X_i \in \mathbb{R}^{C \times T}$ into h groups along the channel dimension. Therefore, each group has C/h channels, and each channel within the same group shares convolutional

weights. The implementation steps are as follows:

$$X_h = \text{Reshape}(X_i) \in \mathbb{R}^{(C/h) \times h \times T} \quad (1)$$

$$X_{dw} = \text{DWConv1D}(X_h, W) \in \mathbb{R}^{(C/h) \times (h*F_1) \times T} \quad (2)$$

$$X_{sse} = \text{Reshape}(X_{dw}) \in \mathbb{R}^{CF_1 \times T}, \quad (3)$$

where DWConv1D denotes 1D depthwise convolution. Additionally, $W \in \mathbb{R}^{(h*F_1) \times 1 \times k}$ is the learnable convolution parameter, and $X_{sse} \in \mathbb{R}^{CF_1 \times T}$ corresponds to the resultant spatial-spectral embedding. It is important to note that W contains $h * F_1$ filters, as each channel uses F_1 filters with kernel size k to generate different spectral characteristics. Compared to depthwise convolution, LightConv reduces the number of parameters by a factor of C/h .

Moreover, by setting different h values and arranging electrode channels, different brain regions can be easily decoded by different temporal filters, which helps extract more comprehensive information. By setting $h = C$ in Eq. (1), the LightConv is equivalent to a depthwise convolution, where each electrode channel uses different filters to extract spectral features. To reduce the number of parameters and accelerate training, we set h as 1 in this paper. All electrode channels share F_1 temporal filters to produce spatial-spectral embedding X_{sse} , as shown in Figure 1.

3.2.2. Adaptive Spatial-Spectral Fusion

An EEG equipment uses multiple electrodes distributed in different regions of the cerebral cortex to capture neuronal activity in these brain regions. When performing different MI tasks, the EEG signal amplitudes recorded by different electrodes may increase or decrease in specific spectral bands [46]. This phenomenon is known as ERD and ERS. Therefore, it is critical to emphasize the relationship between different spectral features among multiple EEG electrodes. When extracting features in the spatial-spectral dimension, each channel should not be treated as equal, but rather, focus should be placed on areas and spectral bands relevant to the specific MI task. Consequently, leveraging the attention mechanism, we design a Spatial-Spectral Attention (SSA) to extract effective spatial-spectral information for all electrodes.

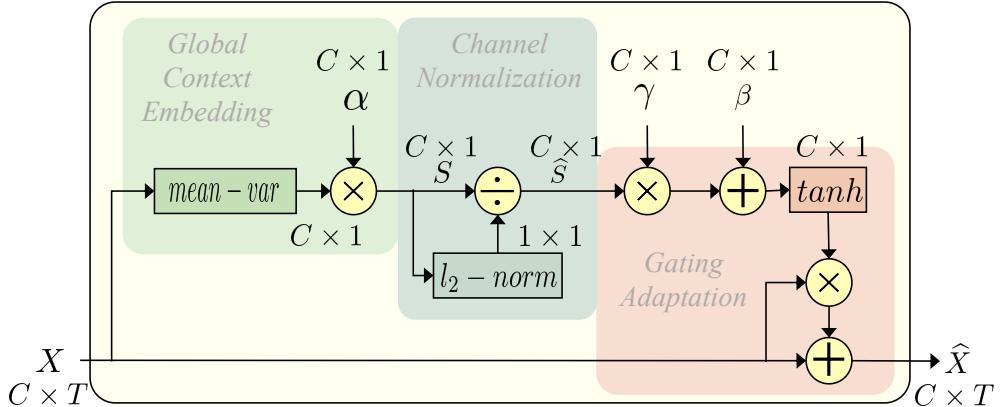


Figure 2: The structure of the proposed spatial-spectral attention.

Inspired by the gated channel transformation [47], our spatial-spectral attention consists of three parts: global context embedding, channel normalization, and gating adaptation, as shown in Figure 2. For an input $X \in \mathbb{R}^{C \times T}$, global context embedding employs a mean-var operation to aggregate temporal information from each channel. This involves calculating the variance within each 1-second window, followed by averaging them. Subsequently, α is responsible for controlling the weight of each channel:

$$s = \alpha \cdot \text{mean-var}(X). \quad (4)$$

Then, we use a channel normalization component to model channel relations:

$$\hat{s} = \frac{\sqrt{C}s}{\|\mathbf{s}\|_2} = \frac{\sqrt{C}s}{[(\sum_{c=1}^C s^2) + \epsilon]^{\frac{1}{2}}}, \quad (5)$$

where ϵ is a small constant to avoid the problem of derivation at the zero point. The gating weight and bias, γ and β are responsible for adjusting the scale of the input feature channel-wise:

$$\text{Attention} = 1 + \tanh(\gamma\hat{s} + \beta) \quad (6)$$

$$\hat{X} = X \cdot \text{Attention}. \quad (7)$$

The scale of each channel of $X_{sse} \in \mathbb{R}^{CF_1 \times T}$ output by the SSE module will be adjusted by the corresponding attention weight. Additionally, SSA leverages global temporal information to model channel relationships and modulate feature maps on the channel-wise level. Therefore, we can effectively fuse the weighted spatial-spectral features using a simple 1D pointwise convolution. As shown in Figure 1, the pointwise convolution uses F_2 filters to simultaneously fuse spectral features of all electrodes to get $X_{assf} \in \mathbb{R}^{F_2 \times T}$.

3.2.3. Multi-scale Variance Pooling

It is crucial to acquire long-term dependencies and global temporal information for EEG decoding. Transformer-based [22, 20, 21, 23] models can capture global information well by using self-attention mechanism, but they have a large number of parameters and high computational complexity. In fact, under the constrained EEG training data, it is difficult for the transformer-based model to achieve optimal performance as in the computer vision field. Therefore, it is necessary to design a new method for EEG decoding that can extract long-term dependencies information.

Metaformer [48] has proposed that using a simple pooling layer in place of self-attention in transformers can also perform well. Therefore, we consider using a pooling layer with a large kernel to extract the global temporal information from the EEG signals. Inspired by [34] and considering that various classes of MI differ in their spectral power (ERD/ERS), a variance operation which represents the spectral power in the given time series becomes a more suitable option for EEG temporal characterization. In order to achieve this, we first design a 1D variance pooling layer, VarPool, which is more compatible with the neural network architecture. For a time series signal $x \in \mathbb{R}^t$, the relationship between its variance Dx and mean Ex is as follows:

$$\begin{aligned} Dx &= E(x - Ex)^2 \\ &= E(x^2) + (Ex)^2 - 2 * (Ex)^2 \\ &= E(x^2) - (Ex)^2. \end{aligned} \quad (8)$$

Therefore, for the EEG representation $X \in \mathbb{R}^{C \times T}$, we can utilize average pooling to calculate variance pooling, as

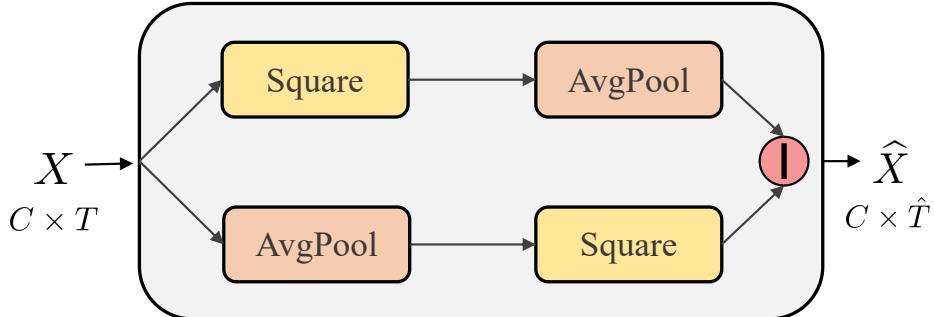


Figure 3: An illustration of the proposed VarPool layer.

shown in Figure 3:

$$\text{VarPool}(X)_{k,s} = \text{AvgPool}(X^2)_{k,s} - \text{AvgPool}(X)_{k,s}^2, \quad (9)$$

where k and s represent the specified window length and sliding step size. For an input $X \in \mathbb{R}^{C \times T}$, the VarPool layer slides along the time dimension to calculate the variance within each window to obtain the output $\hat{X} \in \mathbb{R}^{C \times \hat{T}}$:

$$\hat{T} = \left\lceil \frac{T + 2 \times \text{padding} - (k - 1) - 1}{s} + 1 \right\rceil. \quad (10)$$

Furthermore, to extract multi-scale long-term temporal features, we design the Multi-scale Variance Pooling layer. Specifically, the output X_{assf} of the ASSF module is split into three groups along the channel dimension. VarPool layers with different large kernel sizes (i.e., 50, 100, and 200) are used for each group to extract temporal features. Then, the outputs of the three groups are flattened and concatenated to obtain the final feature vector z_i . It is noteworthy that our MVP module contains no trainable parameters and is computationally efficient.

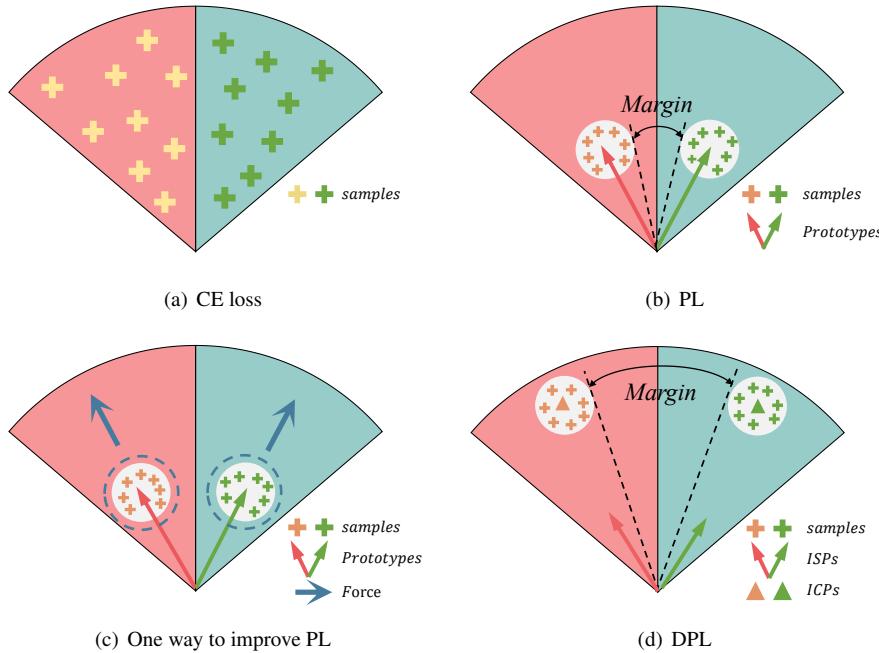


Figure 4: An illustration of the feature space distribution for CE loss, PL, and our proposed DPL.

3.3. Dual Prototype Learning

In classification tasks, existing methods input the feature vector z into a multi-layer perceptron (MLP) to obtain classification results, and optimize the parameters of the neural network using CE loss. Recent studies [21, 11] indicate that CE loss may be lacking in the effectiveness of reducing intra-class variation, especially when considering the non-stationarity of EEG signals. As shown in Figure 4 (a), CE loss only optimizes samples towards the decision boundary of the corresponding class, resulting in a loose distribution of sample features. When applying PL to classification, the first step is to assign a prototype to each class. A classification loss optimizes the sample features to be closest to its corresponding prototypes for classification. Additionally, a prototype loss is used to further push the sample features towards its corresponding prototypes, which can increase intra-class compactness as shown in Figure 4 (b), while also acting as a form of regularization to prevent model overfitting.

Although PL has been widely applied in the field of computer vision, its potential in EEG-MI decoding has been scarcely researched. PL methods focus on utilizing prototype loss to increase intra-class compactness, thereby implicitly enhancing inter-class distance to form a margin, as illustrated in Figure 4 (b). Benefiting from larger

margins, the PL methods outperform CE loss in both general classification tasks [35, 40] and few-shot learning [39]. Therefore, in this paper, we are dedicated to further increasing inter-class margins based on the PL method, in order to enhance the model’s generalization capability in MI decoding tasks with small samples. A natural idea is to extend the clustered features along the direction of their corresponding prototypes, as shown in Figure 4 (c).

We achieve this goal using Dual Prototype Learning, ultimately obtaining a larger inter-class margin as shown in Figure 4 (d).

Specifically, we develop two prototypes for each class: the Inter-class Separation Prototype (ISP) and the Intra-class Compact Prototype (ICP), aiming to achieve inter-class separation and intra-class compactness, respectively. Based on the ISPs, we utilize softmax and CE loss to achieve inter-class separation:

$$\mathcal{L}_S(s, z) = -\frac{1}{m} \sum_{i=1}^m \log \frac{e^{s_{y_i} \cdot z_i}}{\sum_{j=1}^n e^{s_j \cdot z_i}}, \quad (11)$$

where m is the number of training samples, n is the number of classes, z_i is the feature of the i -th sample, y_i is the corresponding label in range $[1, n]$, s represents the ISPs, and $s_j \in \mathbb{R}^d$ is the ISP of class j . Minimizing \mathcal{L}_S can facilitate the separation of features from different classes, resulting in a feature space similar to Figure 4 (a).

Furthermore, we use intra-class compactness loss to compress the distance between samples belonging to the same class in the feature space, which is defined as:

$$\mathcal{L}_C(c, z) = \sum_{i=1}^m D(z_i, c_{y_i}), \quad (12)$$

where c represents the ICPs, $c_{y_i} \in \mathbb{R}^d$ is the ICP of class y_i , and D is the distance function. To prevent training oscillations and mitigate the influence of outlier samples, we use the Huber loss $\mathcal{L}_\delta(z, c)$ with $\delta = 1$ as the distance function $D(z, c)$, which is defined as:

$$\mathcal{L}_\delta(z, c) = \begin{cases} \frac{1}{2}(z - c)^2 & \text{if } |z - c| \leq \delta \\ \delta|z - c| - \frac{1}{2}\delta^2 & \text{if } |z - c| > \delta \end{cases}. \quad (13)$$

\mathcal{L}_C can represent the compactness of each class’s feature vectors. By minimizing \mathcal{L}_C , we can increase the intra-class compactness so that features of the same class are clustered together like Figure 4 (b).

Previous PL methods use a single prototype for each class, as described in Figure 4 (b). In comparison, we decouple inter-class separation and intra-class compactness by using ISPs and ICPs. On one hand, this decouple enhances the robustness of the training process. On the other hand, it provides the conditions for further increasing the inter-class margins. Specifically, we apply an implicit force and an explicit force to the features to achieve the feature space optimization from Figure 4 (c) to Figure 4 (d).

- **Implicit force.** Due to the softmax’s properties [49], the \mathcal{L}_S tends to increase $s_{y_i} \cdot z_i$ until it converges to a constant value during training. This procedure simultaneously displaces the feature vector z_i and its corresponding ISP s_{y_i} away from the origin of the feature space until convergence. Furthermore, if we constrain the norm of ISPs to a smaller value, i.e., $\|s_i\|_2 \leq S$ (weight-normalization), the feature vectors will be pushed further away from the origin. This constraint can act as an implicit force.
- **Explicit force.** To complement the implicit force, we design a simple loss function, \mathcal{L}_{EF} , to increase the norms of ICPs:

$$\mathcal{L}_{EF}(c) = -\|c\|_2. \quad (14)$$

By minimizing \mathcal{L}_{EF} , the norms of ICPs increase, thereby guiding the features to be pushed away from the origin.

During the training phase, the optimization objective of the proposed DPL is as follows:

$$\begin{aligned} & \text{minimize} && \mathcal{L}_S(s, z) + \lambda \mathcal{L}_C(c, z) + \alpha \mathcal{L}_{EF}(c) \\ & \text{subject to} && \|s_i\|_2 \leq S, \quad \forall i = 1, 2, \dots, n \end{aligned}, \quad (15)$$

where λ and α are the trade-off scalar to balance the three losses, and S is set to 1. During the testing phase, the test sample X_i is classified by calculating the dot product between its feature vector z_i and the ISP for each class:

$$\hat{y}_i = \operatorname{argmax}_j(z_i \cdot s_j), \quad j = 1, 2, \dots, n, \quad (16)$$

where \hat{y}_i denotes the predicted result.

4. Experiments and results

4.1. Evaluation Datasets

To demonstrate the effectiveness of our EDPNet, we evaluate it on two public MI-EEG datasets, namely, BCI Competition IV 2a (BCIC-IV-2a) [50] and BCI Competition IV 2b (BCIC-IV-2b) [51]. Additionally, we use the BCI competition III IVa (BCIC-III-IVa) [52] with fewer training data to further validate the generalization ability of the proposed method.

Dataset I. BCIC-IV-2a provided by Graz University of Technology consists of EEG data from 9 subjects. There were four motor imagery tasks, covering the imagination of moving the left hand, right hand, both feet, and the tongue. Two sessions on different days were collected with 22 Ag/AgCl electrodes at a sampling rate of 250 Hz. One session contained 288 EEG trials, i.e., 72 trials per task. We use [2, 6] seconds of each trial and all 22 electrodes in the experiments.

Dataset II. BCIC-IV-2b provided by Graz University of Technology consists of EEG data from 9 subjects. There were two motor imagery tasks, covering the imagination of moving left and right hand. Five sessions were collected with three bipolar electrodes (C3, Cz, and C4) at a sampling rate of 250 Hz and each session contained 120 trials. We use the [3, 7] seconds of each trial in the experiments.

Dataset III. BCIC-III-IVa, recorded at 100 Hz using 118 electrodes, contains 280 trials per subject and comprises two distinct classes: right hand, and foot. This dataset distinguishes itself from other datasets through its imbalanced division into training and testing trials. The quantity of training trials fluctuates between 28 and 224, varying with the subject (al: 224, aa: 168, av: 84, aw: 56, ay: 28), with the residual trials designated for testing. Each trial lasts 3.5 seconds. To preclude overfitting by reducing the number of data points per trial, we select the three channels shared (C3, Cz, and C4).

As the competition guidelines [51] for BCIC-IV-2a and BCIC-IV-2b datasets, we apply hold-out analysis to evaluate the performance of our EDPNet and all comparison methods. As such, the model is trained and tested completely in different sessions. This evaluation method is more in line with practical application scenarios and can better test the generalization ability of the model. For the BCIC-III-IVa dataset, we follow its official protocol to further validate the advantages of our method on small-sample training datasets.

4.2. Experimental Setups

4.2.1. Experimental Details

In this study, we implement our EDPNet using the PyTorch library, based on Python 3.10 with an Nvidia Geforce 2080Ti GPU. We use the AdamW optimizer with default settings (learning rate = 0.001, weight decay = 0.01) to train the feature extractor of our EDPNet. Additionally, we use another Adam optimizer to optimize ISPs and ICPs, with a learning rate of 0.001 on Datasets I and II, and a learning rate of 0.01 on the small-sample Dataset III. Moreover, for the hyperparameters of the model in Figure 1, on Dataset I and II, we empirically set the kernel size of LightConv as 75, F_1 and F_2 as 9 and 48, and the kernel sizes of different scales of the MVP layer as 50, 100, and 200. On Dataset III, due to the differences in sampling rate, we set the kernel size of LightConv as 50, and the kernel sizes of different scales of the MVP layer as 50, 100, and 150.

To prevent overfitting and reduce the number of epochs needed to train the model, a two-stage training strategy as in [8] is used in this work. Specifically, during the training phase, the training data is split into a training set and a validation set. In the first stage, only the training set is used, and the training is stopped if there is no decrease in the validation set loss for N_e consecutive epochs or reach the maximum training epoch N_1 . During the second training stage, all training data are employed, then continue training for N_2 epochs. Due to the different sizes of the datasets used, we set N_1 , N_e , and N_2 to be 1000, 200, and 300 respectively for Dataset I, and 300, 150, and 200, respectively, for Dataset II. For Dataset III, we set them to be 300, 150, and 150.

Table 1: Classification Accuracy(%) and Kappa Comparisons with SOTA Methods on Dataset I.

Methods	A01	A02	A03	A04	A05	A06	A07	A08	A09	Average	Std	Kappa	p-value
FBCSP [53]	76.00	56.50	81.25	61.00	55.00	45.52	82.75	81.25	70.75	67.75	12.89	0.5700	0.0020
EEGNet [10]	85.76	61.46	88.64	67.01	55.90	52.08	89.58	83.33	79.51	74.50	13.85	0.6600	0.0020
TS-SEFFNet [25]	82.29	49.79	87.57	71.74	70.83	<u>63.75</u>	82.92	81.53	81.94	75.17	11.32	0.6630	0.0020
LMDA-Net [16]	83.90	60.30	88.10	<u>78.20</u>	56.20	57.20	88.40	82.70	84.30	75.40	12.78	0.6700	0.0020
Basenet-SE [26]	81.60	52.08	90.28	73.96	76.39	62.85	86.81	80.56	79.51	76.00	11.22	0.6794	0.0020
M-FANet [27]	86.81	75.00	91.67	73.61	76.39	61.46	85.76	75.69	87.17	79.28	<u>8.84</u>	0.7259	0.0137
ATCNet [22]	86.81	68.40	92.01	73.61	<u>78.82</u>	62.15	86.46	<u>87.15</u>	83.33	<u>79.86</u>	9.37	<u>0.7312</u>	0.0020
Conformer[23]	<u>87.85</u>	54.86	86.46	76.04	58.33	59.72	<u>89.58</u>	83.33	81.25	75.27	13.06	0.6702	0.0020
FBMSNet [11]	<u>87.85</u>	66.32	<u>92.36</u>	76.74	72.57	62.15	80.21	86.46	<u>87.85</u>	79.17	9.91	0.7235	0.0020
EDPNet	89.58	71.88	93.06	82.64	81.25	70.14	89.93	89.24	89.24	84.11	7.83	0.7881	-

Best performances are highlighted in bold, while the second-best with underlined.

Table 2: Classification Accuracy(%) and Kappa Comparisons with SOTA Methods on Dataset II.

Methods	B01	B02	B03	B04	B05	B06	B07	B08	B09	Average	Std	Kappa	p-value
FBCSP [53]	70.00	60.36	60.94	97.50	93.12	80.63	78.13	92.50	86.88	80.01	13.06	0.6000	0.0059
EEGNet [10]	71.50	58.65	81.12	96.25	86.23	77.88	85.12	91.10	80.15	80.89	10.43	0.6321	0.0020
TS-SEFFNet [25]	72.81	65.71	75.75	96.25	91.25	85.00	88.63	91.87	82.18	83.27	9.51	0.6637	0.0020
LMDA-Net [16]	<u>75.80</u>	63.20	65.20	<u>97.30</u>	94.30	84.50	82.40	92.90	87.00	82.51	12.40	0.6500	0.0039
Basenet-SE [26]	72.50	<u>67.86</u>	<u>81.13</u>	96.86	93.44	84.69	88.75	93.44	84.69	84.82	<u>9.20</u>	0.6918	0.0057
ATCNet [22]	72.50	67.64	80.31	95.94	<u>96.06</u>	88.12	<u>86.88</u>	89.69	90.94	<u>85.34</u>	9.38	<u>0.7068</u>	0.0645
Conformer [23]	74.56	57.00	62.50	97.01	92.36	83.44	85.00	93.44	87.19	81.39	13.16	0.6265	0.0086
FBMSNet [11]	71.30	55.20	80.55	97.15	95.00	84.66	85.23	91.10	87.13	83.04	12.25	0.6692	0.0039
EDPNet	77.50	68.93	82.81	96.88	96.25	<u>87.50</u>	89.06	93.44	<u>87.50</u>	86.65	8.58	0.7330	-

Best performances are highlighted in bold, while the second-best with underlined.

4.2.2. Performance Metrics

In the experiments, the classification accuracy (ACC) and the Cohen's kappa coefficient (Kappa) are used as two metrics for performance evaluation. The mathematical formula of Cohen's kappa coefficient is defined as follows:

$$Kappa = \frac{P_0 - P_e}{1 - P_e}, \quad (17)$$

where P_0 represents the classification accuracy of the model and P_e represents the expected consistency level. Nevertheless, a one-sided Wilcoxon signed-rank test is used to verify the significance of improvement.

4.3. Overall Performance Comparison

We conduct extensive experiments and compare our method with numerous SOTA approaches across three public datasets. Table 1 displays the classification performance of all methods on Dataset I. Our EDPNet method achieves the highest average accuracy of 84.11% and the highest average kappa value of 0.7881. Moreover, Our method achieves a level of significance with $p < 0.05$ compared to all benchmark methods. The results demonstrate that the classification accuracy of our proposed EDPNet is not only 16.36% higher than the competition champion solution FBCSP ($p < 0.01$) but also significantly surpasses the classic EEGNet by 9.61% ($p < 0.01$). The four latest attention-based methods all apply the attention mechanism to deep feature dimensions. In contrast, our EDPNet utilizes lightweight attention in the spatial-spectral dimension. Consequently, our approach is more interpretable and has an accuracy of 4.83% higher than the best attention-based method M-FANet. Compared to transformer-based methods, we utilize a non-parametric and computationally efficient MVP module to extract long-term temporal features. The results demonstrate that our method achieves accuracy improvements of 4.25% and 8.84% compared to ATCNet and Conformer, respectively.

Table 3: Classification Accuracy(%) and Kappa Comparisons with SOTA Methods on Dataset III.

Methods	al	aa	av	aw	ay	Average	Kappa
	224/256	168/112	84/196	56/224	28/252		
EEG-Net [10]	100	68.75	58.16	79.46	51.59	71.59	0.4331
LMDA-Net [16]	100	70.13	61.33	78.94	52.11	72.50	0.4567
Basenet+SE [26]	100	79.46	<u>64.80</u>	75.89	<u>64.68</u>	<u>76.97</u>	<u>0.5374</u>
ATCnet [22]	100	75.00	61.22	79.02	53.57	73.76	0.4750
FBMSNet [11]	100	<u>82.14</u>	57.14	<u>82.04</u>	59.33	76.13	0.5270
EDPNet	100	88.39	70.41	83.48	67.86	82.03	0.6426

Best performances are highlighted in bold, while the second-best with underlined.

Compared to FBMSNet, our EDPNet not only increases intra-class compactness but also further enlarges inter-class margins, displaying an accuracy improvement of 4.94%.

The experimental results on Dataset II are presented in Table 2, which shows similar results to those on Dataset I. Our EDPNet also achieves the highest average accuracy of 86.65% with the smallest standard deviation and the highest average kappa value of 0.7330. And our method demonstrates significant advantages compared to most of the comparison methods. Notably, on Datasets I and II, our method achieves the best or second-best accuracy for nearly all subjects. Particularly, on Dataset I, for subjects A02 and A06 where other methods do not perform well, our method achieves an accuracy above 70%. As suggested in [11], a BCI system with > 70% binary classification accuracy is generally considered to be usable for healthy subjects and stroke patients. This demonstrates the potential of our EDPNet for MI-based BCI applications.

Moreover, on the smaller training Dataset III, we reproduce several SOTA methods suitable for comparison on this dataset. As shown in Table 3, the number of training samples for the 5 subjects in Dataset III decreases from 224 to 28. Our EDPNet achieves an average recognition accuracy of 82.03% and a kappa value of 0.6426, which are 5.06% and 0.1052 higher than the second-best method, respectively. Especially for the subject "ay" with only 28 training samples, our method still achieves a recognition accuracy of 67.86%. This experimental result demonstrates the superior generalization ability of our method in small-sample EEG decoding tasks.

4.4. Ablation Study

The significant improvement of our EDPNet can be attributed to three novel designs: the Adaptive-Spatial-Spectral fusion module, the Multi-scale Variance Pooling module, and the Dual Prototype Learning approach. To further analyze the impact of these three modules on model performance, we conduct ablation experiments on Datasets I and II. Four models, named Model1, Model2, Model3 and Model4, are utilized, which represent four scenarios as follows:

- **Model1** The model is realized by removing the ASSF module and adopting a depthwise convolution used in EEGNetto fuse the information between EEG electrodes.
- **Model2** This model is implemented by using a single kernel size (100) variance pooling layer, to verify the importance of multi-scale temporal information.
- **Model3** This model removes the DPL module and uses CE loss to optimize parameters.
- **Model4** This model removes the DPL module and uses the PL method [40] to optimize parameters.

Figure 5 shows the ablation data of the accuracy for each subject on Dataset I. It can be clearly seen that the ASSF module brings a 4.83% average accuracy improvement for Model1. This is because the ASSF module uses attention mechanisms to highlight specific spectral features of EEG electrodes related to the specific MI task and more effectively fuses spatial-spectral information. Similarly, the MVP can bring an average accuracy improvement of 2.98% for Model2, as it better adapts to changes in the length of MI-related activity segments between different

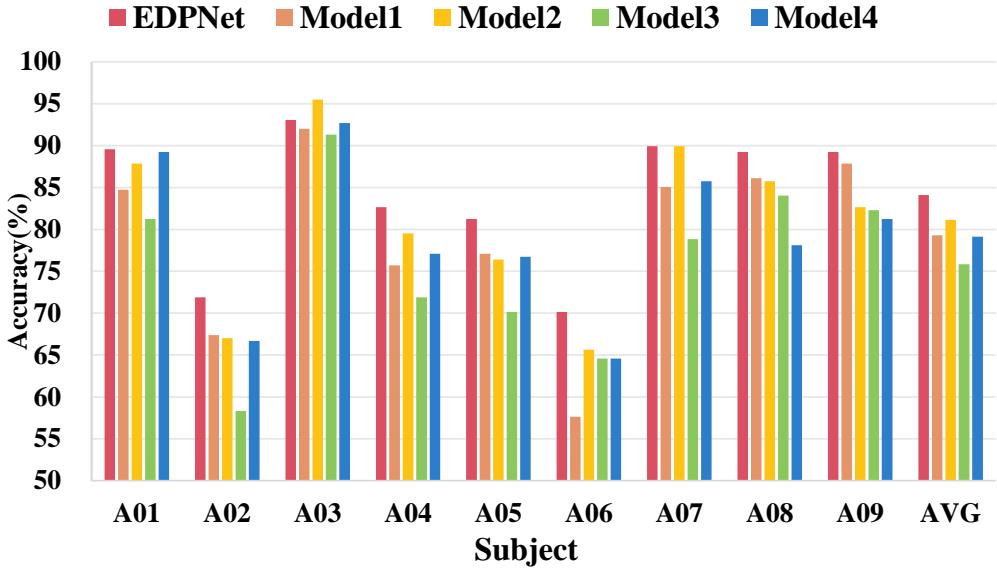


Figure 5: The accuracy comparison of each subject in Dataset I.

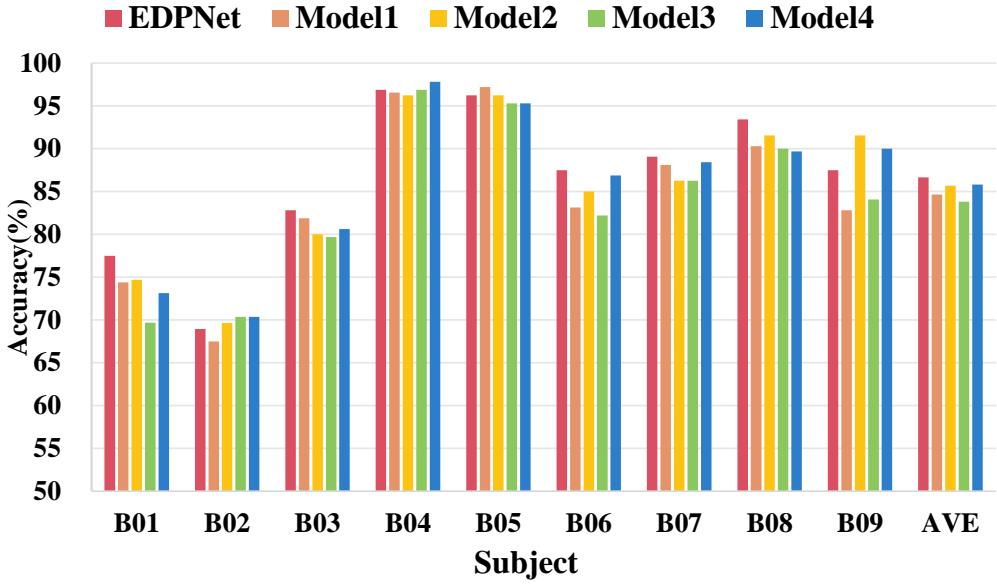


Figure 6: The accuracy comparison of each subject in Dataset II.

subjects. Employing dual prototype learning yields the most substantial enhancements. Compared with Model3, it not only results in an 8.26% average accuracy improvement, but also results in consistent improvement on each subject. Our DPL approach, when compared to CE loss, not only increases intra-class compactness but also further enlarges inter-class margins. This greatly enhances the model’s generalization capability and recognition performance. Moreover, our EDPNet has improved the accuracy by 4.98% compared to Model4. This demonstrates that our DPL method has effectively improved upon the classical PL methods. A similar result is also observed on Dataset II. As shown in Figure 6, on Dataset II, our EDPNet achieves accuracy improvements of 2.00%, 0.96%, 2.82%, and 0.85% compared to Model1, Model2, Model3, and Model4, respectively.

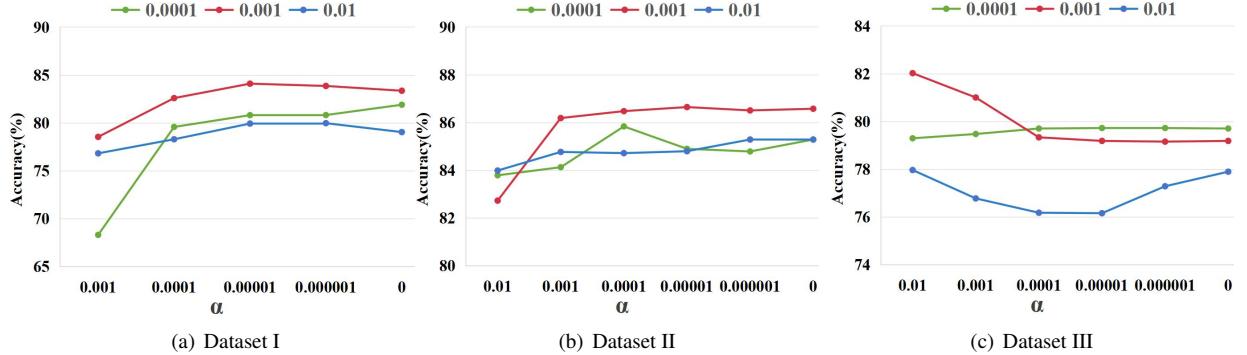


Figure 7: The accuracy of EDPNet across various settings of λ and α on three datasets.

4.5. Parameter Sensitivity

Our EDPNet employs a combination of \mathcal{L}_s , \mathcal{L}_c and \mathcal{L}_{EF} as the final loss function, as shown in Eq. (15). While the \mathcal{L}_s loss aims to minimize misclassification of subject movement intent, the \mathcal{L}_c loss minimizes the sum of the embedded space distance of samples in a class to its center, making the samples belonging to the same class compact in the feature space. And \mathcal{L}_{EF} provides an explicit force, pushing the features away from the origin of the feature space to achieve larger inter-class margins. The λ and α are used to balance the impact of these three losses. To evaluate the influence of the λ and α , an empirical investigation compares the performance of EDPNet across various settings on all three datasets.

As shown in Figure 7, $\lambda = 0.001$ is the most suitable value across all three datasets. When λ is increased to 0.01 or decreased to 0.0001, there is a noticeable decrease in accuracy on Datasets I and II. When λ is fixed at 0.001 and α varies between 0 and 0.001, the accuracies are consistently high and reach the best performance at $\alpha = 0.00001$ on Dataset I and II. It is worth noting that even when $\alpha = 0$, the accuracies on Datasets I and II remain high. This indicates that our DPL can automatically learn larger inter-class margins relying solely on the implicit force. In contrast, on Dataset III, the best performance is achieved when α is increased to 0.01. This implies that increasing the \mathcal{L}_{EF} can further improve classification accuracy on datasets with fewer training samples.

In summary, our EDPNet is relatively robust to the values of the hyperparameters λ and α . When $\lambda = 0.001$ and α is set to a small value (i.e., $\alpha < 0.001$), EDPNet can achieve good performance. If the amount of training data is very limited, further increasing α can be considered.

5. Further Analysis

5.1. Effect of Adaptive Spatial-Spectral Fusion

The key to our ASSF module lies in using the SSA to model the relationships between EEG electrodes. The signal amplitude of specific spectral bands of different EEG electrodes may increase or decrease when performing different MI tasks (such as imagining the hand movement and the foot movement). Our SSA exploits this phenomenon to help the model focus on it for classification. SSA generates adaptive attention weights in the spatial-spectral dimension based on the input EEG representations to re-weight the EEG representations. This enables the model to focus more on spatial-spectral features relevant to the current task.

To further verify the role of the SSA mechanism in imagining movements of different body parts, we use the t-SNE [54] method to visualize the attention vectors defined in Eq. (6) when performing different tasks. As shown in Figure 8(a), there are distinct differences in the distribution of attention vectors when A03 performs MI of the hand and MI of others on Dataset I. Similarly, when the subject "al" in Dataset III performing MI tasks of the hand and foot, the distribution of attention vectors also shows clear boundaries and clusters. This fully demonstrates the effectiveness and interpretability of our SSA mechanism.

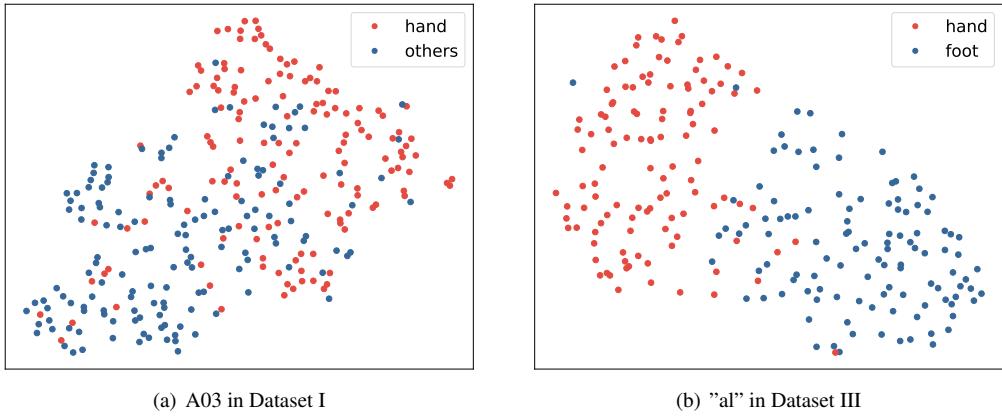


Figure 8: The distribution of attention vectors when performing different MI tasks on two datasets. All attention vectors are mapped to the 2D space using the t-SNE method.

5.2. Effect of Multi-scale Variance Pooling

Our MVP module innovatively uses a pooling layer with a large kernel size to extract long-term temporal information. Moreover, utilizing the crucial prior knowledge of spectral power in EEG signals, we design a variance pooling layer. This integrates the EEG prior into the architecture design of the neural network. Most importantly, compared to transformer-based models, our MVP module has no learnable parameters and is computationally efficient. The ablation experiments in Figure 5 and Figure 6 demonstrate that using only a single kernel size (100) for variance pooling achieves an accuracy of 81.13% and 85.69% on Datasets I and II, respectively. This result is superior to the comparison methods in Table 1 and Table 2.

Moreover, within one trial, the start point and duration of the actual MI period showing the appropriate ERS and ERD pattern can be different from trial to trial [20]. This phenomenon is more significant among trials between different subjects. In order to adapt to these differences between trials and extract more discriminative temporal information, we group the EEG representations along the channel dimension and use variance pooling with different kernel sizes to extract multi-scale temporal information. As shown in Figure 9, a smaller kernel size of 50 performs better on subjects A02, A04, and A09, while a larger kernel size of 200 performs better on subjects A05, A07, and A08. Altogether, the MVP integrates information from different scales and achieves the best overall performance across all subjects.

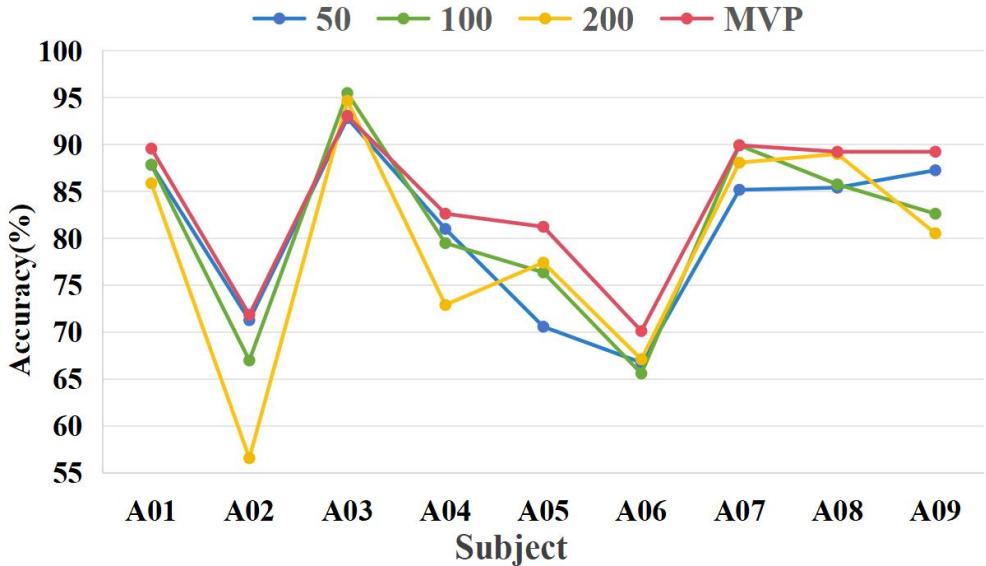


Figure 9: Comparison of the accuracy of single-scale variance pooling and MVP.

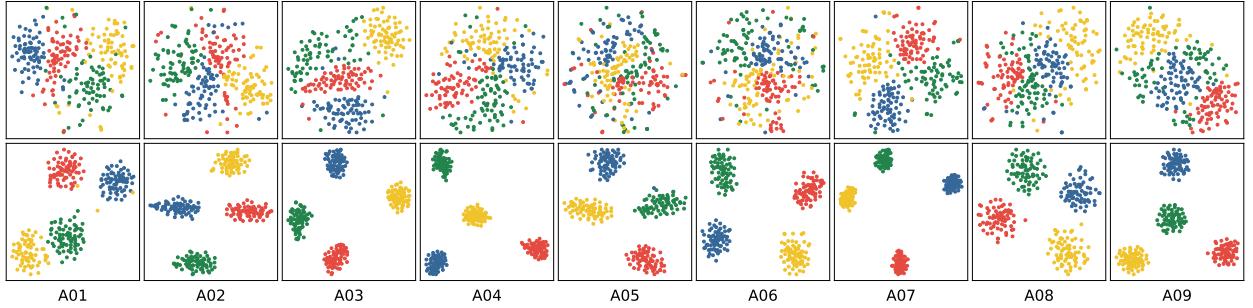


Figure 10: t-SNE visualization of the feature distribution for each subject trained on Dataset I using CE loss (first row) and DPL (second row). The points with different colors denote features from different classes.

5.3. Effect of Dual Prototype Learning

Our EDPNet is the first to apply prototype learning methods to MI-EEG decoding. Furthermore, we decouple inter-class separation and intra-class compactness by using two prototypes, ISP and ICP, for each class. Figure 5 and Figure 6 demonstrate that our DPL method significantly improves the recognition accuracy of MI tasks compared to CE loss. To further explain the effectiveness of DPL, we visualize the distribution of feature vectors z using the t-SNE method. Figure 10 displays the feature distribution of all subjects on Dataset I under both CE loss and DPL optimization approaches. It is apparent that our DPL method achieves greater intra-class compactness and larger inter-class margins compared to CE loss. Therefore, our DPL significantly enhances the model’s generalization ability. This also intuitively explains why our DPL achieves better classification accuracy.

To verify that our DPL pushes the features further away compared to the PL method, thereby achieving larger inter-class margins, we visualize the feature norm distribution of our DPL method and the PL method. Figure 11 shows the L2 norm distribution of deep features trained with DPL and PL on Dataset I. It can be clearly seen that across all subjects, the feature norms trained using DPL are statistically larger than those trained using PL. This realizes the feature space optimization process from Figure 4 (b) to Figure 4 (d), demonstrating the superiority of DPL.

5.4. Computational Expenses

As BCI systems typically operate in online or closed-loop mode on devices with limited computational resources [26], it is crucial to examine the computational expense of new algorithms. Table 4 displays the preprocessing methods, the classification accuracy, model parameters, and Floating Point Operations (FLOPs) for Dataset I. LDMA-Net needs to use Euclidean alignment (EA) to achieve high recognition performance, but EA is not suitable for real-time testing. FBMSNet and M-FANet require time-consuming multi-narrow-band band-pass filtering. In comparison, our EDPNet does not require any preprocessing steps and achieves the highest recognition accuracy with the lowest FLOPs. This suggests that our model optimally balances the accuracy and speed of MI-EEG decoding.

Table 4: Comparisons with SOTA Methods in the Computational Expenses and Recognition Performance on Dataset I.

Methods	Preprocessing	Acc(%)	Kappa	Parameters(k)	FLOPs(M)
LDMA-Net [16]	EA & BP	75.40	0.6700	3.71	50.38
FBMSNet [11]	MBP	79.17	0.7235	16.23	99.95
M-FANet [27]	MBP	79.28	0.7259	4.08	23.39
Conformer [23]	BP	75.27	0.6702	789.57	63.86
ADCNet [22]	no preprocessing	79.86	0.7312	113.73	29.81
EDPNet	no preprocessing	84.11	0.7881	15.21	9.65

BP: band-pass filtering, MBP: multi-narrow-band band-pass filtering.

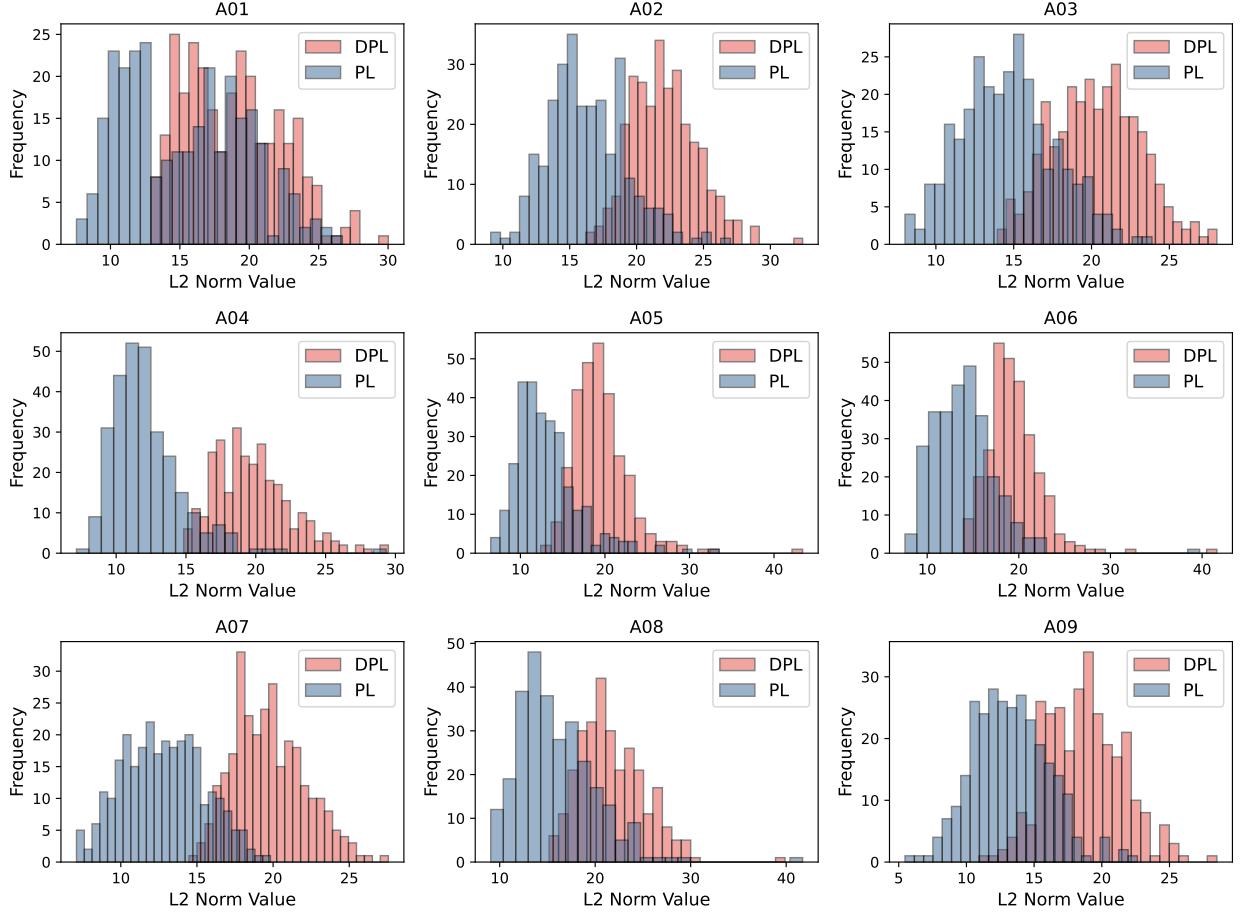


Figure 11: Histogram of the L2 norm distribution of features for each subject trained on Dataset I using PL and DPL, respectively.

Moreover, the number of parameters for our model is much less than transformer-based methods, and is on par with lightweight CNN methods.

5.5. Limitation and Future Work

Although the proposed EDPNet addresses major challenges in EEG-MI decoding and achieves excellent performance, there are some limitations in our current work. First, incorporating more prior knowledge into neural network design is worth exploring, such as considering the mirror distribution of EEG electrodes and the functional partitioning of the brain [24, 55]. Although the LightConv proposed in the SSE module could potentially leverage this prior knowledge, we did not further explore this aspect as it is not the focus of this work. Second, the potential of the brain-inspired DPL framework remains to be fully explored. By decoupling inter-class separation and intra-class compactness, we simply constrain the prototype and feature distribution to achieve superior performance. Nonetheless, more effective and discriminative loss functions deserve further investigation. Finally, EDPNet has only undergone offline testing on public datasets and has not yet been validated in an online BCI environment. In the future, we will continue to enhance EDPNet based on these avenues to achieve good performance in online BCI applications.

6. Conclusion

In this paper, we propose a lightweight and efficient dual prototype network for MI-EEG decoding. Based on neurophysiological priors and EEG data characteristics, we design the ASSF module and MVP module to extract

high discriminative features from EEG signals. The ASSF module utilizes a lightweight SSA mechanism to model the relationship between EEG electrodes for the extraction of powerful spatial-spectral features. Then, the MVP module is used to capture multi-scale long-term temporal features. Moreover, inspired by the recognition mechanism of the human brain, we propose a novel DPL approach to explicitly increase intra-class compactness and inter-class margins in the feature space. The DPL enhances the model's generalization capability, thereby helping to alleviate the limited sample issue. We conduct extensive experiments on three public datasets, and the results confirm that our method surpasses other SOTA methods. The proposed EDPNet holds promising potential for MI-based BCI applications due to its remarkable performance combined with low computational expenses.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was partially supported by OYMotion Technologies.

References

- [1] X. Gao, D. Xu, M. Cheng, S. Gao, A bci-based environmental controller for the motion-disabled, *IEEE Transactions on neural systems and rehabilitation engineering* 11 (2) (2003) 137–140.
- [2] R. Abiri, S. Borhani, E. W. Sellers, Y. Jiang, X. Zhao, A comprehensive review of eeg-based brain–computer interface paradigms, *Journal of neural engineering* 16 (1) (2019) 011001.
- [3] K. Sakai, K. Goto, J. Tanabe, K. Amimoto, K. Kumai, H. Kamio, Y. Ikeda, Effects of visual-motor illusion on functional connectivity during motor imagery, *Experimental Brain Research* 239 (7) (2021) 2261–2271.
- [4] P. D. E. Banique, E. C. Stanyer, M. Awais, A. Alazmani, A. E. Jackson, M. A. Mon-Williams, F. Mushtaq, R. J. Holt, Brain–computer interface robotics for hand rehabilitation after stroke: A systematic review, *Journal of neuroengineering and rehabilitation* 18 (2021) 1–25.
- [5] K. K. Ang, C. Guan, Eeg-based strategies to detect motor imagery for control and rehabilitation, *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 25 (4) (2016) 392–401.
- [6] J. Long, Y. Li, H. Wang, T. Yu, J. Pan, F. Li, A hybrid brain computer interface to control the direction and speed of a simulated or real wheelchair, *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 20 (5) (2012) 720–729.
- [7] B. J. Edelman, J. Meng, D. Suma, C. Zurn, E. Nagarajan, B. S. Baxter, C. C. Cline, B. He, Noninvasive neuroimaging enhances continuous neural tracking for robotic device control, *Science robotics* 4 (31) (2019) eaaw6844.
- [8] R. T. Schirrmeyer, J. T. Springenberg, L. D. J. Fiederer, M. Glassetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, T. Ball, Deep learning with convolutional neural networks for eeg decoding and visualization, *Human brain mapping* 38 (11) (2017) 5391–5420.
- [9] S. Sakhavi, C. Guan, S. Yan, Learning temporal information for brain-computer interface using convolutional neural networks, *IEEE transactions on neural networks and learning systems* 29 (11) (2018) 5619–5629.
- [10] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, B. J. Lance, Eegnet: a compact convolutional neural network for eeg-based brain–computer interfaces, *Journal of neural engineering* 15 (5) (2018) 056013.
- [11] K. Liu, M. Yang, Z. Yu, G. Wang, W. Wu, Fbmsnet: A filter-bank multi-scale convolutional neural network for eeg-based motor imagery decoding, *IEEE Transactions on Biomedical Engineering* 70 (2) (2022) 436–445.
- [12] X. Gu, F. Deligianni, J. Han, X. Liu, W. Chen, G.-Z. Yang, B. Lo, Beyond supervised learning for pervasive healthcare, *IEEE Reviews in Biomedical Engineering* (2023).
- [13] X. Zhang, L. Yao, X. Wang, J. Monaghan, D. Mcalpine, Y. Zhang, A survey on deep learning-based non-invasive brain signals: recent advances and new frontiers, *Journal of neural engineering* 18 (3) (2021) 031002.
- [14] S. Li, H. Wu, L. Ding, D. Wu, Meta-learning for fast and privacy-preserving source knowledge transfer of eeg-based bcis, *IEEE Computational Intelligence Magazine* 17 (4) (2022) 16–26.
- [15] J. Han, X. Gu, G.-Z. Yang, B. Lo, Noise-factorized disentangled representation learning for generalizable motor imagery eeg classification, *IEEE Journal of Biomedical and Health Informatics* (2023).
- [16] Z. Miao, M. Zhao, X. Zhang, D. Ming, Lmda-net: A lightweight multi-dimensional attention network for general eeg-based brain-computer interfaces and interpretability, *NeuroImage* 276 (2023) 120209.
- [17] W. Tao, Z. Wang, C. M. Wong, Z. Jia, C. Li, X. Chen, C. P. Chen, F. Wan, Adfcnn: Attention-based dual-scale fusion convolutional neural network for motor imagery brain-computer interface, *IEEE Transactions on Neural Systems and Rehabilitation Engineering* (2023).
- [18] X. Tang, C. Yang, X. Sun, M. Zou, H. Wang, Motor imagery eeg decoding based on multi-scale hybrid networks and feature enhancement, *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 31 (2023) 1208–1218.
- [19] X. Liu, S. Xiong, X. Wang, T. Liang, H. Wang, X. Liu, A compact multi-branch 1d convolutional neural network for eeg-based motor imagery classification, *Biomedical Signal Processing and Control* 81 (2023) 104456.
- [20] P. Deny, S. Cheon, H. Son, K. W. Choi, Hierarchical transformer for motor imagery-based brain computer interface, *IEEE Journal of Biomedical and Health Informatics* (2023).

- [21] H.-J. Ahn, D.-H. Lee, J.-H. Jeong, S.-W. Lee, Multiscale convolutional transformer for eeg classification of mental imagery in different modalities, *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 31 (2022) 646–656.
- [22] H. Altaheri, G. Muhammad, M. Alsulaiman, Physics-informed attention temporal convolutional network for eeg-based motor imagery classification, *IEEE transactions on industrial informatics* 19 (2) (2022) 2249–2258.
- [23] Y. Song, Q. Zheng, B. Liu, X. Gao, Eeg conformer: Convolutional transformer for eeg decoding and visualization, *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 31 (2022) 710–719.
- [24] J. Zhang, K. Li, B. Yang, X. Han, Local and global convolutional transformer-based motor imagery eeg classification, *Frontiers in Neuroscience* 17 (2023) 1219988.
- [25] Y. Li, L. Guo, Y. Liu, J. Liu, F. Meng, A temporal-spectral-based squeeze-and-excitation feature fusion network for motor imagery eeg decoding, *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 29 (2021) 1534–1545.
- [26] M. Wimpff, L. Gizzii, J. Zerfowski, B. Yang, Eeg motor imagery decoding: A framework for comparative analysis with channel attention mechanisms, *Journal of Neural Engineering* 21 (3) (2024) 036020.
- [27] Y. Qin, B. Yang, S. Ke, P. Liu, F. Rong, X. Xia, M-fanet: Multi-feature attention convolutional neural network for motor imagery decoding, *IEEE Transactions on Neural Systems and Rehabilitation Engineering* (2024).
- [28] K. Zhang, N. Robinson, S.-W. Lee, C. Guan, Adaptive transfer learning for eeg motor imagery classification with deep convolutional neural network, *Neural Networks* 136 (2021) 1–10.
- [29] S. Pérez-Velasco, E. Santamaría-Vázquez, V. Martínez-Cagigal, D. Marcos-Martínez, R. Hornero, Eegsym: Overcoming inter-subject variability in motor imagery based bcis with deep learning, *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 30 (2022) 1766–1775.
- [30] H. W. Ng, C. Guan, Subject-independent meta-learning framework towards optimal training of eeg-based classifiers, *Neural Networks* 172 (2024) 106108.
- [31] K. Yin, E. Y. Lim, S.-W. Lee, Gitgan: Generative inter-subject transfer for eeg motor imagery analysis, *Pattern Recognition* 146 (2024) 110015.
- [32] D. Borra, S. Fantozzi, E. Magosso, Interpretable and lightweight convolutional neural network for eeg decoding: Application to movement execution and imagination, *Neural Networks* 129 (2020) 55–74.
- [33] T. Hanakawa, I. Immisch, K. Toma, M. A. Dimyan, P. Van Gelderen, M. Hallett, Functional properties of brain areas associated with motor execution and imagery, *Journal of neurophysiology* 89 (2) (2003) 989–1002.
- [34] R. Mane, N. Robinson, A. P. Vinod, S.-W. Lee, C. Guan, A multi-view cnn with novel variance layer for motor imagery brain computer interface, in: 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), IEEE, 2020, pp. 2950–2953.
- [35] H.-M. Yang, X.-Y. Zhang, F. Yin, C.-L. Liu, Robust classification with convolutional prototype learning, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 3474–3482.
- [36] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, Q. Hu, Eca-net: Efficient channel attention for deep convolutional neural networks, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 11534–11542.
- [37] J. Hu, L. Shen, G. Sun, Squeeze-and-excitation networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 7132–7141.
- [38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, *Advances in neural information processing systems* 30 (2017).
- [39] J. Snell, K. Swersky, R. Zemel, Prototypical networks for few-shot learning, *Advances in neural information processing systems* 30 (2017).
- [40] H.-M. Yang, X.-Y. Zhang, F. Yin, Q. Yang, C.-L. Liu, Convolutional prototype network for open set recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44 (5) (2020) 2358–2370.
- [41] F. C. Borlino, S. Bucci, T. Tommasi, Contrastive learning for cross-domain open world recognition, in: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2022, pp. 10133–10140.
- [42] Z. Xia, P. Wang, G. Dong, H. Liu, Adversarial kinetic prototype framework for open set recognition, *IEEE Transactions on Neural Networks and Learning Systems* (2023).
- [43] B. Zhang, X. Li, Y. Ye, Z. Huang, L. Zhang, Prototype completion with primitive knowledge for few-shot learning, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 3754–3762.
- [44] F. Zhou, P. Wang, L. Zhang, W. Wei, Y. Zhang, Revisiting prototypical network for cross domain few-shot learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 20061–20070.
- [45] F. Wu, A. Fan, A. Baevski, Y. N. Dauphin, M. Auli, Pay less attention with lightweight and dynamic convolutions, *arXiv preprint arXiv:1901.10430* (2019).
- [46] H. Altaheri, G. Muhammad, M. Alsulaiman, S. U. Amin, G. A. Altuwaijri, W. Abdul, M. A. Bencherif, M. Faisal, Deep learning techniques for classification of electroencephalogram (eeg) motor imagery (mi) signals: A review, *Neural Computing and Applications* 35 (20) (2023) 14681–14722.
- [47] Z. Yang, L. Zhu, Y. Wu, Y. Yang, Gated channel transformation for visual recognition, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 11794–11803.
- [48] W. Yu, C. Si, P. Zhou, M. Luo, Y. Zhou, J. Feng, S. Yan, X. Wang, Metaformer baselines for vision, *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023).
- [49] F. Wang, X. Xiang, J. Cheng, A. L. Yuille, Normface: L2 hypersphere embedding for face verification, in: Proceedings of the 25th ACM international conference on Multimedia, 2017, pp. 1041–1049.
- [50] M. Tangermann, K.-R. Müller, A. Aertsen, N. Birbaumer, C. Braun, C. Brunner, R. Leeb, C. Mehring, K. J. Miller, G. R. Müller-Putz, et al., Review of the bci competition iv, *Frontiers in neuroscience* 6 (2012) 55.
- [51] K. K. Ang, Z. Y. Chin, C. Wang, C. Guan, H. Zhang, Filter bank common spatial pattern algorithm on bci competition iv datasets 2a and 2b, *Frontiers in neuroscience* 6 (2012) 21002.
- [52] B. Blankertz, K.-R. Muller, D. J. Krusienski, G. Schalk, J. R. Wolpaw, A. Schlogl, G. Pfurtscheller, J. R. Millan, M. Schroder, N. Birbaumer,

- The bci competition iii: Validating alternative approaches to actual bci problems, *IEEE transactions on neural systems and rehabilitation engineering* 14 (2) (2006) 153–159.
- [53] K. K. Ang, Z. Y. Chin, H. Zhang, C. Guan, Filter bank common spatial pattern (fbcsp) in brain-computer interface, in: 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence), IEEE, 2008, pp. 2390–2397.
 - [54] L. Van der Maaten, G. Hinton, Visualizing data using t-sne., *Journal of machine learning research* 9 (11) (2008).
 - [55] J. Luo, Y. Wang, S. Xia, N. Lu, X. Ren, Z. Shi, X. Hei, A shallow mirror transformer for subject-independent motor imagery bci, *Computers in Biology and Medicine* 164 (2023) 107254.