

EEG-DBNet: A Dual-Branch Network for Temporal-Spectral Decoding in Motor-Imagery Brain-Computer Interfaces[★]

Xicheng Lou^a, Xinwei Li^b, Hongying Meng^c, Jun Hu^d, Meili Xu^d, Yue Zhao^a, Jiazhang Yang^e, Zhangyong Li^{b,*}

^a*School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China*

^b*Research Center of Biomedical Engineering, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China*

^c*Department of Electrical, Brunel University London, London, UB8 3PH, United Kingdom*

^d*Department of Neurology, Southwest Hospital, Army Medical University, Chongqing, 400038, China*

^e*Yongchuan Women and Children Hospital, Chongqing, 402160, China*

Abstract

Motor imagery electroencephalogram (EEG)-based brain-computer interfaces (BCIs) offer significant advantages for individuals with restricted limb mobility. However, challenges such as low signal-to-noise ratio and limited spatial resolution impede accurate feature extraction from EEG signals, thereby affecting the classification accuracy of different actions. To address these challenges, this study proposes an end-to-end dual-branch network (EEG-DBNet) that decodes the temporal and spectral sequences of EEG signals in parallel through two distinct network branches. Each branch comprises a local convolutional block and a global convolutional block. The local convolutional block transforms the source signal from the temporal-spatial domain to the temporal-spectral domain. By varying the number of filters and convolution kernel sizes, the local convolutional blocks in different branches adjust the length of their respective dimension sequences. Different types of pooling layers are then employed to emphasize the features of various dimension sequences, setting the stage for subsequent global feature extraction. The global convolution block splits and reconstructs the feature of the signal sequence processed by the local convolution block in the same branch, and further extracts features through the dilated causal convolutional neural networks. Finally, the outputs from the two branches are concatenated, and signal classification is completed via a fully connected layer. Our proposed method achieves classification accuracies of 85.84% and 91.60% on the BCI Competition IV-2a and BCI Competition IV-2b datasets, respectively, surpassing existing state-of-the-art models. The source code is available at <https://github.com/xicheng105/EEG-DBNet>.

Keywords:

electroencephalogram (EEG), motor imagery (MI), brain-computer interfaces (BCIs), neural networks.

1. Introduction

As a direct pathway for information exchange between the human brain and computers, brain-computer interfaces (BCIs) translate neural activity into commands for controlling external devices [1]. Typically, BCIs extract features from electroencephalogram (EEG) signals, measured by electrodes placed on the scalp, and convert them into outputs to control devices such as wheelchairs and displays. Motor imagery EEG (MI-EEG) is a spontaneous electrical potential generated when subjects imagine movements without actual movement. MI-EEG-based BCIs have wide applications in motor control [2], neural rehabilitation [3], and specialized environmental operations [4].

However, decoding MI-EEG poses several challenges. The limited number of electrodes in acquisition devices results in low spatial resolution. The prevalent use of non-invasive methods, driven by safety and portability, introduces isolation between electrodes and the brain, leading to a low signal-to-noise ratio (SNR). The low spatial resolution makes methods that transform signal sequences into topological maps highly susceptible to the influence of the number of electrodes, with more electrodes generally improving accuracy [5]. Meanwhile, the low SNR increases the likelihood of models mistaking noise for features during training, necessitating robust feature extraction capabilities. Consequently, data preprocessing is crucial for traditional machine learning-based EEG decoding methods, including linear discriminant analysis (LDA) [6], support vector machine (SVM) [7], naive Bayesian classifier (NBC) [8], and K-nearest neighbor (KNN) [9]. Preprocessing techniques such as filtering [10], transformation [11], artifact removal [12], and electrode selection [13] require specialized knowledge and significant computational resources.

Deep neural networks offer an integrated approach to data preprocessing, providing an end-to-end solution for EEG signal decoding and becoming the mainstream method in the field.

[★]This work was supported by the National Natural Science Foundation of China [grant number 62171073, 62311530103, 62106032 and 62236002]; “Chunhui Plan” Collaborative Research Project of the Ministry of Education, China [grant number HZKY20220209]; Natural Science Foundation of Chongqing, China [grant number CSTB2023NSCQ-LZX0064]; Chongqing Scientific Research Innovation Project for Postgraduate Students [grant number CYB23241]; and the Doctoral Training Program of Chongqing University of Posts and Telecommunications [grant number BYJS202317].

*Corresponding author

Email address: lizy@cqupt.edu.cn (Zhangyong Li)

Convolutional neural networks (CNNs), known for their feature extraction capabilities in image processing, have been applied to EEG decoding. Schirrneister et al. [14] proposed two models with different complexities: Deep ConvNet and Shallow ConvNet. Inspired by the filter bank common spatial pattern (FBCSP) [15], these models augmented the pseudo-spectral dimension of raw signals, simulated spectral characteristics via channel-wise convolution, and compressed the spatial dimension. Lawhern et al. [16] enhanced the Shallow ConvNet by introducing depth-wise separable convolution from MobileNet [17], creating EEGNet, a compact and efficient network that has become a benchmark in EEG decoding models, with numerous subsequent models building upon its structure for refinement. Chen et al. [18] proposed EEG-NeX, which builds upon EEGNet by increasing the number of convolutional layers to extract more temporal feature maps. They replaced separable convolutions with traditional convolutions to enhance feature extraction accuracy and designed an inverse bottleneck structure for the convolutional channels to achieve feature reconstruction. Additionally, they introduced dilated convolutions [19] to improve the network's receptive field. EEG-NeX demonstrated a significant improvement in classification accuracy compared to EEGNet. Ingolfsson et al. [20] proposed EEG-TCNet, which constructed dilated causal convolutional neural networks (DC-CNNs) using temporal convolutions [21] and dilated convolutions to enhance feature extraction based on EEGNet. Since dilated convolutions enhance the receptive field by increasing network depth, EEG-TCNet incorporated a residual structure [22] to avoid network degradation issues caused by deeper networks. Altaheri et al. [23] introduced ATCNet, which further builds on EEG-TCNet by incorporating a sliding window for data augmentation and leveraging a multi-head attention mechanism [24] to strengthen the extraction of global signal features. ATCNet significantly improved the classification accuracy of signals in the BCI Competition IV-2a dataset.

Some scholars believe that the scale of EEG signal samples can no longer match the depth of the model's network. They tend to improve model performance by increasing the number of network branches. Salami et al. [25] introduced the Inception structure [26], utilizing different convolutional kernels to construct three branches based on EEGNet, thereby increasing the diversity of features extracted from EEG signals. Altuwaijri et al. [27] further enhanced the model by incorporating squeeze-and-excitation (SE) modules [28] to reconstruct the relationships between signals within the three-branch network. Tao et al. [29] developed two network branches using separable convolutions and traditional convolutions, respectively, to extract different spectral and spatial information. The branching structure enables models to simultaneously construct multiple signal decoding paradigms, thereby improving model performance through multi-dimensional signal decoding. Since the original EEG signals are spatiotemporal two-dimensional signals lacking a spectral dimension, some researchers have constructed the spectral dimension using traditional signal processing techniques instead of convolution. Mane et al. [30] increased the dimensionality of the source EEG through band-pass filtering, extracted features via convolution, and aggregated

temporal information through a variance layer. Li et al. [31] extracted multi-frequency band information from EEG using wavelet transform and reconstructed this information using SE modules. Zhi et al. [32] expanded the spectral domain channels through filtering and subsequently extracted signal features in three dimensions: spatial, frequency, and time-frequency domains. Hsu and Cheng [33] employed wavelet transform to reconstruct signal features, enhancing the discrimination of time-frequency maps.

Significant progress has been made in the decoding techniques for MI-EEG, but further improvements are necessary. First, the advantage of branch networks lies in their ability to decode multiple dimensions of a signal simultaneously. Simply reconstructing multiple temporal sequences using each branch to enhance the extraction of temporal features does not fully leverage the strengths of branch networks. Moreover, generating spectral sequences using traditional signal processing methods will be re-modeled during the decoding process, which is not conducive to branch networks training filters based on temporal information to automatically generate spectral sequences that are adapted for subsequent processing steps. Second, convolution and pooling processes compress the sequence length, serving as a feature extraction process. After compression, each sample point on the sequence represents an aggregation of adjacent samples of the source signal, encapsulating local features of the original sequence. However, the characteristics of different types of MI-EEG are expressed through the entire sequence, meaning that local features alone cannot adequately represent MI-EEG.

This study proposes a dual-branch network called EEG-DBNet, where two network branches parallelly decode the temporal and spectral sequences of the MI-EEG. Each branch consists of a local convolution (LC) block and a global convolution (GC) block. The LC block re-models the signal sequences of different dimensions, extracting local features and providing source signals for the GC block. The GC block extracts global features specific to different dimension sequences. The main contributions of this study are as follows:

1. We propose a novel dual-branch network, EEG-DBNet, which combines parallel multi-dimensional decoding and serial multi-scale feature extraction techniques to improve the classification accuracy of MI-EEG signals.
2. For the branches handling signal sequences of different dimensions, we arranged different numbers of filters and various sizes of convolution kernels in the LC blocks to reconstruct the signal sequences. Additionally, we optimized the feature representation of signal sequences of different dimensions by employing different types of pooling layers. The LC blocks effectively extract local features of the signals. Compared to the single-branch structure of EEGNet [16], there is a significant improvement in the accuracy of signal classification.
3. We enhanced DCCNNs with sequence splitting and feature reconstruction techniques, constructing GC blocks to perform global feature extraction on the sequences reconstructed by LC blocks. Sequence splitting enables DC-

CNNs to simultaneously model multiple feature extraction paradigms; feature reconstruction for each split subsequence strengthens the expression of subsequence features, thereby improving the efficiency of DCCNNs. Introducing GC blocks after LC blocks significantly enhances model performance.

4. Our proposed EEG-DBNet achieved classification accuracies of 85.58% and 91.60% in signal classification experiments on the public datasets BCI Competition IV-2a [34] and BCI Competition IV-2b [35], respectively. These results surpass previous models and achieve state-of-the-art performance.

The remainder of this paper is organized as follows: [section 2](#) details the proposed EEG-DBNet model; [section 3](#) discusses the experiments and their results; [section 4](#) provides the conclusions of this study.

2. Method

The structure of EEG-DBNet is depicted in [Figure 1](#). The network consists of two branches, each comprising a Local Convolutional (LC) block and a Global Convolutional (GC) block. After the signal is processed through these branches, the outputs are concatenated and classified using a fully connected (FC) layer.

The input signal $\mathbf{X}_i \in \mathbb{R}^{C \times T}$ for EEG-DBNet is the standardized original signal $\mathbf{Z}_i \in \mathbb{R}^{C \times T}$, where trial $i \in \{1, 2, \dots, m\}$, T denotes the number of sample points, C is the number of EEG electrodes, and m is the number of trials

$$\mathbf{x}_{\text{train},i}^j = \frac{\mathbf{z}_{\text{train},i}^j - \mu_{\text{train},i}^j}{\sigma_{\text{train},i}^j} \quad (1)$$

$$\mathbf{x}_{\text{test},i}^j = \frac{\mathbf{z}_{\text{test},i}^j - \mu_{\text{train},i}^j}{\sigma_{\text{train},i}^j} \quad (2)$$

Here, $\mathbf{z}_{\text{train},i}^j \in \mathbb{R}^{1 \times T}$ and $\mathbf{z}_{\text{test},i}^j \in \mathbb{R}^{1 \times T}$ are the training and testing samples for electrode $j \in \{1, 2, \dots, C\}$. $\mathbf{x}_{\text{train},i}^j \in \mathbb{R}^{1 \times T}$ and $\mathbf{x}_{\text{test},i}^j \in \mathbb{R}^{1 \times T}$ are the corresponding standardized samples. $\mu_{\text{train},i}^j$ and $\sigma_{\text{train},i}^j$ represent the mean and variance of the training samples, respectively. EEG-DBNet is trained on the set $\{\mathbf{X}_{\text{train},i}, y_i\}_{i=1}^m$, where $y_i \in \{1, 2, \dots, n\}$ is the label of \mathbf{X}_i . The trained model is then used for classification on $\mathbf{X}_{\text{test},i}$.

2.1. Local Convolutional Block

The structure of the LC block in each branch is similar to that of EEGNet [16], as shown in [Figure 2](#). It consists of three convolutional layers: a basic convolutional layer, a depthwise convolutional layer, and a separable convolutional layer, along with two pooling layers. Batch normalization (BN) [36] is applied after the first convolutional layer to accelerate convergence. BN and the exponential linear unit (ELU) [37] are applied after the second and third convolutional layers. The dropout rate [38] is set to 0.3.

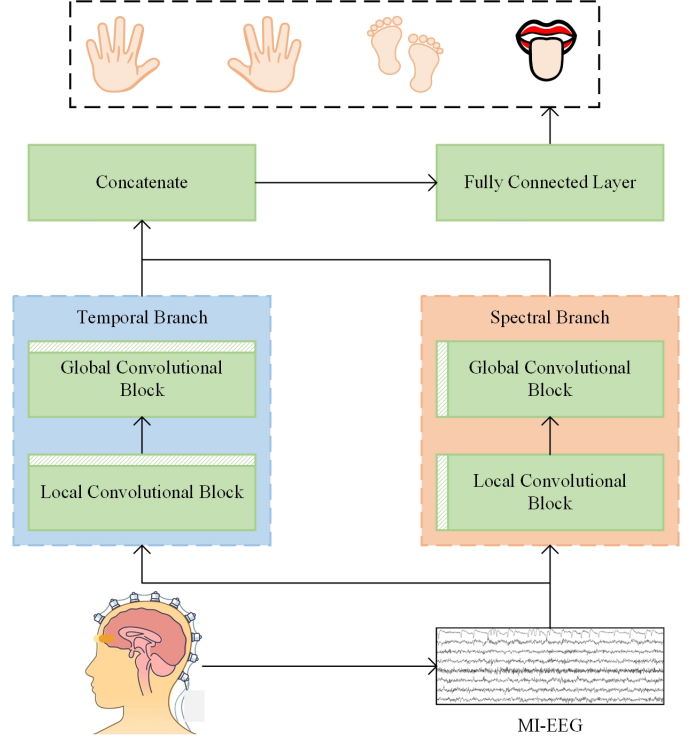


Figure 1: Structure of the proposed EEG-DBNet model.

[Figure 2\(a\)](#) illustrates the LC block structure in the temporal branch. The first convolutional layer applies temporal convolution using \hat{F}_1 filters of size $(1, \hat{K})$, where $\hat{F}_1 = 8$ and \hat{K} is set to one-fifth of the sampling frequency, which is 250Hz in BCI Competition IV-2a and BCI Competition IV-2b. Thus, $\hat{K} = 48$. This temporal convolution extracts features above 5Hz, producing \hat{F}_1 temporal feature maps.

The second convolutional layer performs depthwise convolution with depth $D = 2$ and $\hat{F}_1 \times D$ filters of size $(C, 1)$. This depthwise convolution extracts D spatial features from each temporal feature map, generating a total of $\hat{F}_1 \times D$ feature maps. Next, an average pooling layer of size $(1, \hat{K}/8)$ further integrates the temporal features, reducing the signal's sampling frequency to approximately 42Hz ($250/\hat{K}/8$).

The third convolutional layer employs \hat{F} filters of size $(1, \hat{K}/4)$ to perform separable convolution, where $\hat{F} = \hat{F}_1 \times D$. This separable convolution models the interrelationships of feature maps with a minimal number of parameters and compresses the temporal feature extraction range to approximately 286ms ($1000 \times \hat{K}/4/42$). Finally, another average pooling layer of size $(1, \hat{K}/8)$ reduces the sampling frequency to 7Hz ($42/\hat{K}/8$).

Each \hat{F} corresponds to a temporal sequence at a specific frequency, so sample points at the same position in the time-domain sequences for all \hat{F} form a spectral sequence. The input signal $\mathbf{X}_i \in \mathbb{R}^{C \times T}$ is processed by the temporal branch LC block, resulting in the output $\hat{\mathbf{X}}_{L,i} \in \mathbb{R}^{\hat{F} \times \hat{T}}$. The length of \hat{T} can be determined as:

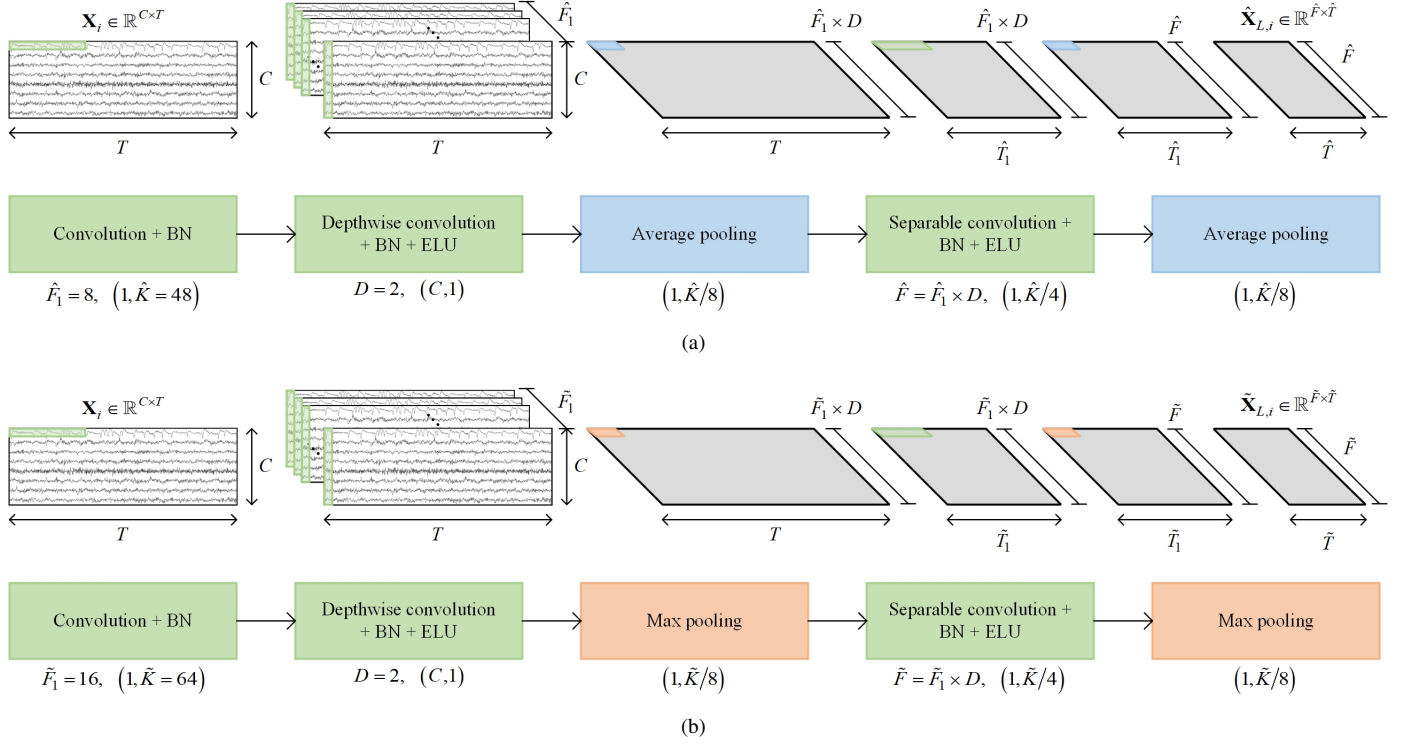


Figure 2: Structure of the local convolutional blocks. (a) local convolutional block in the temporal branch. (b) local convolutional block in the spectral branch.

$$\hat{T} = \left\lfloor \frac{64 \times T}{\hat{K}^2} \right\rfloor \quad (3)$$

The structure of the spectral branch LC block, as shown in Figure 2(b), includes twice the number of filters compared to the temporal branch, specifically $\tilde{F}_1 = 16$, to enhance the feature representation of the spectral sequence. Consequently, the length of the spectral sequence also doubles to $\tilde{F} = 32$. The filter size is increased to a quarter of the sampling frequency, i.e., $\tilde{K} = 64$, which narrows the time-domain range for feature extraction and thereby shortens the length of the temporal sequence \tilde{T} . The length \tilde{T} can be determined as follows:

$$\tilde{T} = \left\lfloor \frac{64 \times T}{\tilde{K}^2} \right\rfloor \quad (4)$$

The output of this branch can be denoted as $\tilde{\mathbf{X}}_{L,i} \in \mathbb{R}^{\tilde{F} \times \tilde{T}}$. This adjustment results in an increased spectral sequence length and a decreased temporal sequence length from the spectral branch, aligning its dimensions closely with those of the output signal from the temporal branch, thereby maintaining the symmetry of the branch network. Moreover, as demonstrated by the experiments in section 3.4, for the spectral sequence, the importance of key sample points in expressing sequence features outweighs the relationships between sequence sample points. Therefore, the spectral branch employs max-pooling layers.

2.2. Global Convolutional Block

The GC block extracts global features from the output of the LC block, as shown in Figure 3. A sliding window (SW) with

a stride of 1 first segments the signal sequences from the corresponding network branch into n subsequences. The temporal branch signal sequence is denoted as \hat{T} , and the length of each subsequence is given by:

$$\hat{l} = \hat{T} - n + 1 \quad (5)$$

Thus, The input $\hat{\mathbf{X}}_{L,i} \in \mathbb{R}^{\hat{F} \times \hat{T}}$ of the GC block in the temporal branch is split into $\hat{\mathbf{X}}_{L,N,i} \in \mathbb{R}^{\hat{F} \times (\hat{T}-n+1)}$, $N \in \{1, 2, \dots, n\}$, as illustrated in Figure 3(a). Feature reconstruction for each subsequence provides diverse samples for subsequent feature mining networks, thereby enriching the mined features. We employ a dimension-reduction version of SE [28] to reconstruct subsequence features. Temporal subsequences first undergo global average pooling to mask distribution information in the spectral dimension:

$$\begin{aligned} \hat{\mathbf{x}}_{av,N,i} &= \hat{f}_{av}(\hat{\mathbf{X}}_{L,N,i}) \\ &= \frac{1}{\hat{F}} \sum_{j=1}^{\hat{F}} \hat{\mathbf{X}}_{L,N,i}(j) \end{aligned} \quad (6)$$

Subsequently, two FC layers are employed to reconstruct features. The first FC layer compresses the sequence length to $\lceil (\hat{T} - n + 1)/16 \rceil$, and the second FC layer restores the sequence length to its original state. A rectified linear unit (ReLU) [39] activation function follows the first FC layer to speed up network convergence, while a sigmoid activation function follows the second FC layer to transform the output to a range between 0 and 1, serving as the weight sequence for sampling points. Finally, the sequence is recalibrated through the Hadamard product:

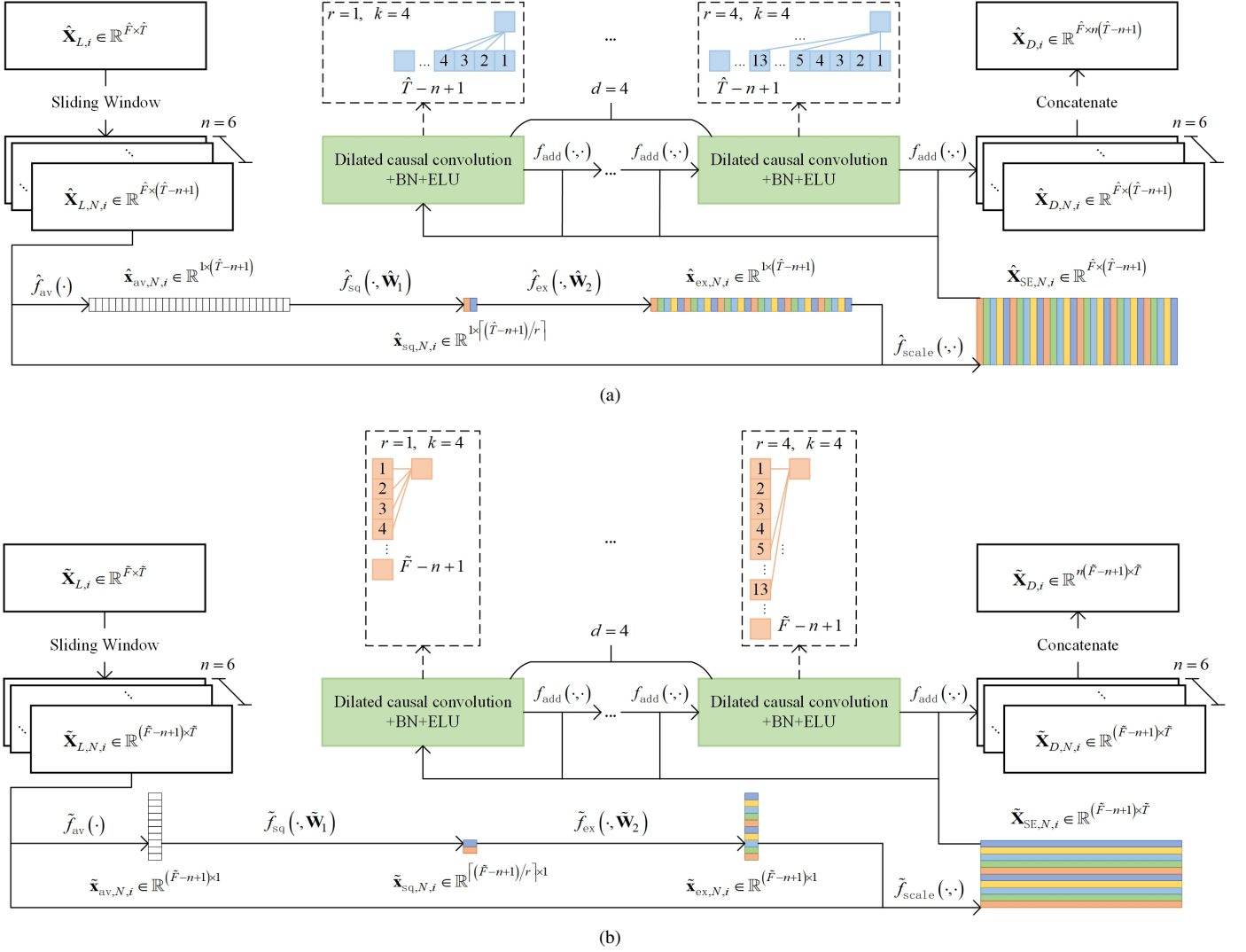


Figure 3: The structure of the global convolutional blocks with six sliding windows, four dilated causal convolution layers, and the kernel size is set to 4. (a) global convolutional block in the temporal branch. (b) global convolutional block in the spectral branch.

$$\begin{aligned} \hat{\mathbf{X}}_{SE,N,i} &= \hat{f}_{scale}(\hat{\mathbf{X}}_{L,N,i}, \hat{f}_{sq}(\hat{f}_{av}(\hat{\mathbf{X}}_{av,N,i}, \hat{\mathbf{W}}_1), \hat{\mathbf{W}}_2)) \\ &= \hat{\mathbf{X}}_{L,N,i} \odot \sigma(\hat{\mathbf{W}}_2 \delta(\hat{\mathbf{W}}_1 \hat{\mathbf{X}}_{av,N,i})) \end{aligned} \quad (7)$$

where $\hat{\mathbf{X}}_{SE,N,i}$ represent the signals after feature reconstruction, \odot denotes the Hadamard product, σ represents the sigmoid activation function, δ denotes the ReLU activation function, and $\hat{\mathbf{W}}_1, \hat{\mathbf{W}}_2$ are parameters of the fully connected layers.

The subsequence $\hat{\mathbf{X}}_{SE,N,i}$ after feature reconstruction undergoes d dilated causal convolutional (DCC) layers, each followed by BN and ELU to extract global features. Let the dilation rate r for each DCC layer be j , indicating the j -th DCC layer, and k be the kernel size. The receptive field R of the DCCNNs can be determined as follows:

$$R_j = R_{j-1} + [(d-1) \times (k-1) + k] - 1 \quad (8)$$

where $R_0 = 1$. To avoid information loss, the receptive field must be larger than the subsequence length, implying that the

hyperparameters of the DCCNNs must satisfy:

$$\begin{aligned} R_j &= R_{j-1} + [(d-1) \times (k-1) + k] - 1 \\ &\geq \hat{T} - n + 1 \end{aligned} \quad (9)$$

Through the experiments in section 3.4, we determined that the optimal values for d and k in both branches are 4 and 4, respectively.

The output of each DCC layer is added to the source subsequence before being fed into the next DCC layer:

$$\begin{aligned} \hat{\mathbf{X}}_{D,N,i} &= f_{add}(\hat{\mathbf{X}}_{SE,N,i}, \hat{\mathbf{X}}_{DCC,j}) \\ &= \zeta(\hat{\mathbf{X}}_{SE,N,i} + \hat{\mathbf{X}}_{DCC,j}) \end{aligned} \quad (10)$$

$$\begin{aligned} \hat{\mathbf{X}}_{DCC,j} &= \hat{f}_{DCC,j}(f_{add}(\hat{\mathbf{X}}_{SE,N,i}, \hat{\mathbf{X}}_{DCC,j-1})) \\ &= \hat{f}_{DCC,j}(\zeta(\hat{\mathbf{X}}_{SE,N,i} + \hat{\mathbf{X}}_{DCC,j-1})) \end{aligned} \quad (11)$$

where $j \in \{2, 3, \dots, d\}$, ζ refers to the ELU function, $\hat{f}_{DCC,j}$ denotes the j -th DCC layer in the temporal branch, and $\hat{\mathbf{X}}_{DCC,j}$

is the out put of this DCC layer. This residual structural is necessary because deep DCCNNs disrupt the sequence, excessively emphasizing backend results, which may strengthen the “patched” features. The output $\hat{\mathbf{X}}_{D,i} \in \mathbb{R}^{\hat{F} \times n(\hat{T}-n+1)}$ of the GC block in the temporal branch is obtained by concatenating n processed subsequences $\hat{\mathbf{X}}_{D,N,i} \in \mathbb{R}^{\hat{F} \times (\hat{T}-n+1)}$, $N \in \{1, 2, \dots, n\}$.

For the spectral branch, the signal sequence is denoted as \tilde{F} . After partitioning using a sliding window, the subsequences can be represented as

$$\tilde{l} = \tilde{F} - n + 1 \quad (12)$$

The input $\tilde{\mathbf{X}}_{L,i} \in \mathbb{R}^{\tilde{F} \times \tilde{T}}$ of the GC block in the spectral branch is thus split into $\tilde{\mathbf{X}}_{L,N,i} \in \mathbb{R}^{(\tilde{F}-n+1) \times \tilde{T}}$, $N \in \{1, 2, \dots, n\}$, as illustrated in Figure 3(b). The spectral subsequence masks the information in the temporal dimension through global average pooling

$$\begin{aligned} \tilde{\mathbf{x}}_{av,N,i} &= \tilde{f}_{av}(\tilde{\mathbf{X}}_{L,N,i}) \\ &= \frac{1}{\tilde{T}} \sum_{j=1}^{\tilde{T}} \tilde{\mathbf{X}}_{L,N,i}(j). \end{aligned} \quad (13)$$

The feature reconstruction of the spectral subsequence is achieved by

$$\begin{aligned} \tilde{\mathbf{X}}_{SE,N,i} &= \tilde{f}_{scale}(\tilde{\mathbf{X}}_{L,N,i}, \tilde{f}_{ex}(\tilde{f}_{sq}(\tilde{\mathbf{x}}_{av,N,i}, \tilde{\mathbf{W}}_1), \tilde{\mathbf{W}}_2)) \\ &= \tilde{\mathbf{X}}_{L,N,i} \odot \sigma(\tilde{\mathbf{W}}_2 \delta(\tilde{\mathbf{W}}_1 \tilde{\mathbf{x}}_{av,N,i})) \end{aligned} \quad (14)$$

where $\tilde{\mathbf{X}}_{SE,N,i}$ represent the signals after feature reconstruction, and $\tilde{\mathbf{W}}_1, \tilde{\mathbf{W}}_2$ are parameters of the fully connected layers. The structure and hyperparameters of the DCCNNs in the spectral branch are identical to those in the temporal branch; the difference is that it processes spectral sequences. Therefore, the receptive field must satisfy

$$\begin{aligned} R_j &= R_{j-1} + [(d-1) \times (k-1) + k] - 1 \\ &\geq \tilde{F} - n + 1. \end{aligned} \quad (15)$$

Combining with (9)

$$R_j \geq \max\{\hat{T} - n + 1, \tilde{F} - n + 1\} \quad (16)$$

The output $\tilde{\mathbf{X}}_{D,i} \in \mathbb{R}^{n(\tilde{F}-n+1) \times \tilde{T}}$ of the GC block in the spectral branch is obtained by concatenating n processed subsequences $\tilde{\mathbf{X}}_{D,N,i} \in \mathbb{R}^{(\tilde{F}-n+1) \times \tilde{T}}$, $N \in \{1, 2, \dots, n\}$, where

$$\begin{aligned} \tilde{\mathbf{X}}_{D,N,i} &= f_{add}(\tilde{\mathbf{X}}_{SE,N,i}, \tilde{\mathbf{X}}_{DCC,j}) \\ &= \zeta(\tilde{\mathbf{X}}_{SE,N,i} + \tilde{\mathbf{X}}_{DCC,j}) \end{aligned} \quad (17)$$

$$\begin{aligned} \tilde{\mathbf{X}}_{DCC,j} &= \tilde{f}_{DCC,j}(f_{add}(\tilde{\mathbf{X}}_{SE,N,i}, \tilde{\mathbf{X}}_{DCC,j-1})) \\ &= \tilde{f}_{DCC,j}(\zeta(\tilde{\mathbf{X}}_{SE,N,i} + \tilde{\mathbf{X}}_{DCC,j-1})) \end{aligned} \quad (18)$$

$\tilde{f}_{DCC,j}$ denotes the j -th DCC layer in the spectral branch.

Vectorize $\tilde{\mathbf{X}}_{D,i}$ and $\tilde{\mathbf{X}}_{D,i}$ to obtain $\hat{\mathbf{x}}_i \in \mathbb{R}^{\hat{F}n(\hat{T}-n+1)}$ and $\tilde{\mathbf{x}}_i \in \mathbb{R}^{n(\tilde{F}-n+1)\tilde{T}}$. Concatenate $\hat{\mathbf{x}}_i$ and $\tilde{\mathbf{x}}_i$, and perform signal classification through an FC layer to obtain the predicted label \check{y}_i

$$\check{y}_i = \sigma(\mathbf{W}(\hat{\mathbf{x}}_i \frown \tilde{\mathbf{x}}_i)) \quad (19)$$

where \frown denotes concatenate, \mathbf{W} is parameters of the FC layer.

3. Experiments

3.1. Datasets

The effectiveness of the proposed EEG-DBNet was validated through extensive experiments on two public datasets: BCI Competition IV-2a [34] and IV-2b [35].

BCI Competition IV-2a consists of two sessions, each comprising EEG data recorded using 22 electrodes collected from 9 subjects. Each subject participated in a 4-class motor imagery EEG experiment, including left hand (L), right hand (R), feet (F), and tongue (T) imagery. Each subject completed 72 trials for each class in each session, resulting in a total of 576 trials per subject. The time segment for each trial is set to [1.5, 6] seconds. In this study, data from the first session serve as the training set, while data from the second session serve as the test set.

BCI Competition IV-2b consists of five sessions, each comprising EEG data recorded using three electrodes collected from 9 subjects. Each subject participated in a 2-class motor imagery EEG experiment, including left-hand (L) and right-hand (R) imagery. In this study, data from the first three sessions serve as the training set, while data from the remaining sessions served as the test set. Each subject completed 400 trials in the training set and 320 trials in the test set, resulting in a total of 720 trials per subject. The time segment for each trial is set to [2.5, 7] seconds.

3.2. Experiment Settings

The proposed EEG-DBNet and other comparative models were trained and tested using the TensorFlow framework, running on the Intel(R) Xeon(R) Gold 6248R CPU @ 3.00GHz and NVIDIA A10 GPU. Both datasets utilized data from all EEG channels, excluding three electrooculography (EOG) channels. All models were trained using the Adam optimizer with a learning rate of 0.0009, a batch size of 64, and cross-entropy loss over 1000 epochs with early stopping set to patience of 300 epochs. The best-performing round out of ten training rounds, based on test results, was selected for evaluation.

The evaluation metrics used were classification accuracy P_a and the kappa value K . The classification accuracy P_a is defined as:

$$P_a = \frac{\sum_{i=1}^n \text{TP}_i / l_i}{n} \quad (20)$$

where TP_i denotes the number of correctly predicted (true positive) samples in class i , l_i is the number of samples in class i , and n is the number of classifications. The kappa value K is defined as:

$$K = \frac{P_a - P}{1 - P} \quad (21)$$

where P indicates the hypothetical probability of chance agreement.

3.3. Performance Comparison

The proposed EEG-DBNet is compared with six state-of-the-art methods in recent years: EEGNet[16], EEG-NeX[18], EEG-ITNet[25], EEG-TCNet[20], MBEEGSE[27],

Table 1: Classification Performance ($P_a(\%)$ and K) on BCI Competition IV-2a for the Proposed EEG-DBNet with Other Models

Subject	EEGNet[16]		EEG-NeX[18]		EEG-ITNet[25]		EEG-TCNet[20]		MBEEGSE[27]		ATCNet[23]		EEG-DBNet	
	P_a	K	P_a	K	P_a	K	P_a	K	P_a	K	P_a	K	P_a	K
A1	89.32	0.8577	85.05	0.8007	86.48	0.8197	85.05	0.8007	90.39	0.8719	88.61	0.8481	90.75	0.8766
A2	63.25	0.5103	71.02	0.6136	67.84	0.5714	64.31	0.5246	65.37	0.5386	73.85	0.6516	74.56	0.6610
A3	94.87	0.9316	96.34	0.9512	95.60	0.9414	93.41	0.9121	95.60	0.9414	97.44	0.9658	97.44	0.9658
A4	67.54	0.5666	85.09	0.8009	74.12	0.6543	70.18	0.6021	75.88	0.6781	81.58	0.7539	82.46	0.7659
A5	75.00	0.6664	81.52	0.7535	77.90	0.7050	77.90	0.7053	77.54	0.6999	80.80	0.7443	82.61	0.7678
A6	60.93	0.4792	63.72	0.5163	64.19	0.5229	62.79	0.5036	61.40	0.4852	73.95	0.6525	75.35	0.6713
A7	89.89	0.8654	90.61	0.8749	90.97	0.8797	88.45	0.8461	93.14	0.9086	93.50	0.9134	91.34	0.8844
A8	87.82	0.8377	83.39	0.7786	83.03	0.7737	82.66	0.7688	88.56	0.8475	90.04	0.8671	88.93	0.8524
A9	81.44	0.7523	84.09	0.7878	85.23	0.8030	84.09	0.7878	85.98	0.8130	89.77	0.8635	89.39	0.8586
St.D.*	11.93	0.1592	9.18	0.1224	9.86	0.1315	10.16	0.1355	11.52	0.1537	7.90	0.1053	7.23	0.0963
Mean	78.90	0.7186	82.32	0.7642	80.60	0.7412	78.76	0.7168	81.54	0.7538	85.50	0.8067	85.87	0.8115

* Standard deviation.

Table 2: Classification Performance ($P_a(\%)$ and K) on BCI Competition IV-2b for the Proposed EEG-DBNet with Other Models

Subject	EEGNet[16]		EEG-NeX[18]		EEG-ITNet[25]		EEG-TCNet[20]		MBEEGSE[27]		ATCNet[23]		EEG-DBNet	
	P_a	K	P_a	K	P_a	K	P_a	K	P_a	K	P_a	K	P_a	K
B1	79.82	0.5970	80.70	0.6127	82.46	0.6498	80.26	0.6074	82.02	0.6420	81.58	0.6320	85.09	0.7019
B2	71.43	0.4282	77.14	0.5428	75.51	0.5102	75.51	0.5101	71.43	0.4282	73.06	0.4607	75.10	0.5018
B3	90.00	0.7998	92.61	0.8515	90.00	0.7979	90.87	0.8168	88.26	0.7652	90.87	0.8172	91.74	0.8339
B4	98.37	0.9674	99.02	0.9805	98.70	0.9739	98.70	0.9739	98.70	0.9739	99.02	0.9805	98.37	0.9674
B5	99.63	0.9927	99.63	0.9927	98.53	0.9707	98.53	0.9707	98.53	0.9707	100.00	1.0000	99.27	0.9853
B6	93.63	0.8723	92.03	0.8406	90.44	0.8083	89.24	0.7842	92.03	0.8403	90.04	0.8009	92.43	0.8483
B7	95.69	0.9138	94.83	0.8966	94.40	0.8879	94.83	0.8966	94.83	0.8966	96.55	0.9310	95.26	0.9052
B8	95.22	0.9043	96.09	0.9217	96.52	0.9303	94.35	0.8869	95.22	0.9042	95.22	0.9043	96.96	0.9390
B9	89.80	0.7959	89.80	0.7958	88.16	0.7629	89.80	0.7958	91.02	0.8203	90.20	0.8039	90.20	0.8038
St.D.*	9.22	0.0721	7.12	0.0556	8.04	0.0630	8.14	0.0624	9.46	0.0805	8.92	0.0670	8.35	0.0661
Mean	90.40	0.8079	91.32	0.8261	90.52	0.8102	90.23	0.8047	90.23	0.8046	90.73	0.8145	91.60	0.8318

* Standard deviation.

and ATCNet[23]. EEGNet, a classic method for decoding MI-EEG, is widely regarded as a benchmark. EEG-NeX is an improved version of EEGNet. The other models share structural similarities with the proposed method. EEG-ITNet and MBEEGSE utilize a branched architecture, while EEG-TCNet and ATCNet include modules for extracting global features. We reproduced all these methods on the BCI Competition IV-2a and IV-2b datasets based on the model parameters described in the original papers.

Table 1 and Table 2 present the experimental results for each model on the BCI Competition IV-2a and IV-2b datasets, respectively. EEG-DBNet achieved the best results on the BCI Competition IV-2a dataset. ATCNet’s performance was close to that of EEG-DBNet, with a significant increase in accuracy compared to other methods. Notably, EEG-DBNet also had the lowest standard deviation, indicating its superior generalization capability. Similarly, EEG-DBNet achieved the best results on the BCI Competition IV-2b dataset. Given that the BCI Competition IV-2b dataset is derived from a binary classification task, which is simpler compared to the four-class classification

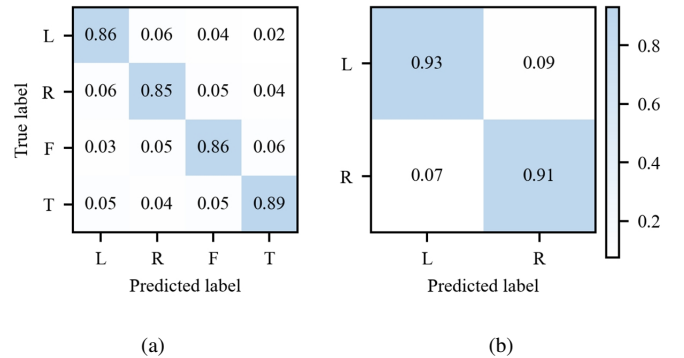


Figure 4: Confusion matrices of EEG-DBNet. “L” stands for left hand, “R” stands for right hand, “F” stands for feet, and “T” stands for tongue. (a) BCI Competition IV-2a. (b) BCI Competition IV-2b.

in BCI Competition IV-2a, all methods attained relatively high classification accuracy. In this dataset, EEG-NeX performed

comparably to EEG-DBNet and demonstrated the best generalization. EEG-DBNet outperformed other methods by a margin of one percentage point. Overall, the proposed EEG-DBNet achieved optimal results on both datasets, demonstrating its excellent EEG decoding capabilities. The confusion matrix for EEG-DBNet is shown in Figure 4.

3.4. Ablation Experiments

The effectiveness of each module of EEG-DBNet was validated through ablation experiments on the BCI Competition IV-2a. As shown in Table 3, the dual-branch network, which employs average pooling for temporal sequences and max pooling for spectral sequences (LC block), improves accuracy by 3.5% over the single-branch network, achieving 82.43%. This indicates that the dual-branch network has superior feature extraction capabilities, which are achieved by constructing diversified CNNs. The introduction of DCCNNs further enhances accuracy by 0.35%, reaching 82.78%, demonstrating that DCCNNs serve as a valuable supplement to CNNs. The key difference between DCCNNs and CNNs is that DCCNNs increase the receptive field by adding network depth, thereby enhancing the network’s global feature extraction ability.

Table 3: Contribution of EEG-DBNet blocks for BCI Competition IV-2a

Block	$\bar{P}_a(\%)^*$	\bar{K}^*
Single-branch (EEGNet[16])	78.90	0.7186
Dual-branch (LC ¹)	82.43	0.7656
LC ¹ +GC without SE and SW	82.78	0.7703
LC ¹ +GC without SE	84.39	0.7918
LC ¹ +GC without SW	83.61	0.7814
LC ² +GC	83.16	0.7754
LC ³ +GC	82.85	0.7713
LC ⁴ +GC	81.90	0.7587
LC ¹ +GC (EEG-DBNet)	85.87	0.8115

* Average of nine subjects.

¹ Average pooling for temporal branch and max pooling for spectral branch.

² Average pooling for both branches.

³ Max pooling for both branches.

⁴ Max pooling for temporal branch and average pooling for spectral branch.

Splitting the sequence into six subsequences and then extracting features from these using DCCNNs improves the model’s classification accuracy by 1.61%, reaching 84.39%. This suggests that sequence splitting enables DCCNNs to construct different feature extraction paradigms and increase feature diversity. By reconstructing the features of the sequence and then extracting features using DCCNNs, the model’s classification accuracy improves by 0.83%, reaching 83.61%. This indicates that remodeling the relationships between sequence samples helps DCCNNs more accurately extract signal features. When both sequence splitting and feature reconstruction are used simultaneously, DCCNNs processing the reconstructed subsequences can improve the model’s classification accuracy by 3.09%, surpassing the sum of their individual effects (1.61%+0.83% = 2.44%), reaching 85.87%. This demonstrates that reconstructed subsequences can enhance the efficiency of DCCNNs in constructing different feature extraction

paradigms. The introduction of the GC block based on DCCNNs significantly improves model performance, increasing accuracy by 3.44% over the dual-branch network with only the LC block, indicating that the GC block and LC block provide substantial complementary effects.

Furthermore, Table 3 shows that the model performance is highly sensitive to the type of pooling layer used for different signal dimensions. If both network branches utilize average pooling, the accuracy of the model decreases by 2.71% compared to EEG-DBNet, reaching 83.16%. If both network branches use max pooling, the accuracy decreases by 3.02%, dropping to 82.85%. When max pooling is used in the temporal branch and average pooling in the spectral branch, accuracy drops by 3.97%, reaching 81.90%, which is even lower than the performance of the network composed solely of LC. Therefore, we deduce that for spectral sequences, specific sample points better reflect the sequence characteristics than the relationships between multiple sample points. Conversely, for temporal sequences, the relationships between sample points are more important than individual points. As a result, average pooling is more suitable for temporal sequences, while max pooling is more appropriate for spectral sequences.

3.5. Parameter Sensitivity

The GC block has four hyperparameters: the stride s of the sliding window, the number of sliding windows n , the kernel size k of the dilated convolution, and the number of dilated convolution layers d . The first step of the GC block is to split the sequence, where s must be determined to segment the sequence into n parts based on its length. We conducted 36 experiments on the BCI Competition IV-2a with $s \in \{1, 2, 3\}$ and $n \in \{2, 3, 4, 5\}$, matching the corresponding k and d according to the constraints of (16). We recorded the average P_a (\bar{P}_a) and average K (\bar{K}) across nine subjects to evaluate the impact of different s values on the GC block. The experimental results are illustrated in Figure 5, where the results of the 36 experiments are divided into three groups based on different s values. It can be observed that as the s increases, the classification accuracy of the model on the signal decreases. This is likely due to the larger stride accentuating the differences in the features of each subsequence. Therefore, we set $s = 1$, and then choose $n \in \{2, 3, 4, 5, 6, 7, 8\}$ to select all d and k that satisfy the constraints of (16) for our experiments. The results, as shown in Figure 6, indicate that $d = 4$ and $k = 4$ meet the constraint conditions for all values of n , while $d = 3$ and $k = 5$ only satisfy the constraints when $n = 8$. The model exhibits its worst and best accuracy both at $d = 4$ and $k = 4$. Specifically, the model performs worst at $d = 4, k = 4, n = 3$ with $\bar{P}_a = 83.42\%$ and $\bar{K} = 0.7789$, and it performs best at $d = 4, k = 4, n = 6$ with $\bar{P}_a = 85.87\%$ and $\bar{K} = 0.8115$. Consequently, we set the four hyperparameters of the GC block to $s = 1, d = 4, k = 4$, and $n = 6$.

4. Conclusion and Discussion

This study proposes EEG-DBNet, a dual-branch network structure designed to parallelly decode the temporal and spec-

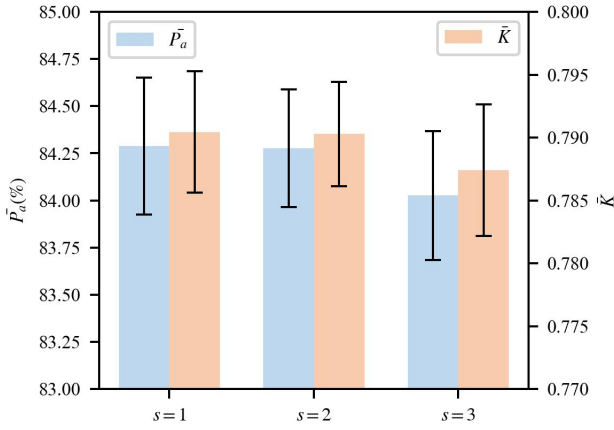


Figure 5: Classification performance ($P_a(\%)$ and K) on BCI Competition IV-2a for different values of s .

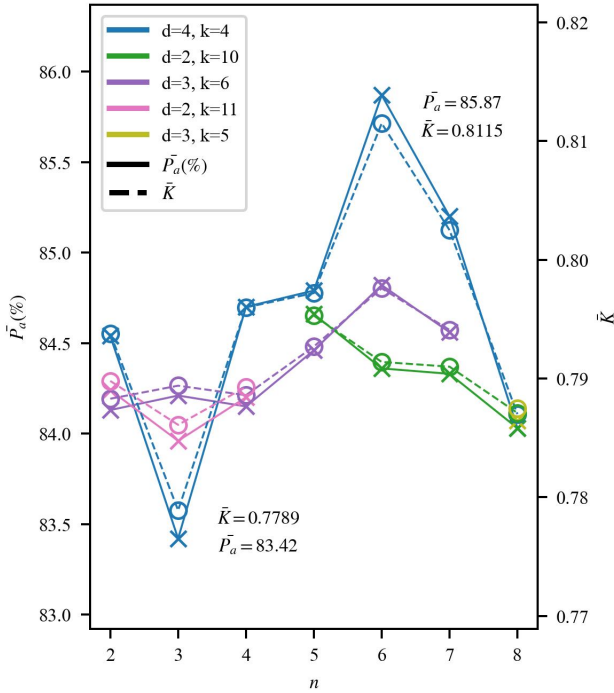


Figure 6: Classification performance ($P_a(\%)$ and K) on BCI Competition IV-2a for various hyperparameters ($s = 1$).

tral sequences of MI-EEG, accurately discerning subjects' motor intentions. The temporal sequences alone provide limited signal features; thus, a multi-dimensional exploration of these features is crucial for enhancing model performance. The parallel dual-branch network is essential for this decoding approach. If temporal and spectral sequences are decoded sequentially, the model accuracy drops below 81%, highlighting the equal importance of both feature types for decoding MI-EEG.

The primary function of the LC block is to remodel sequence sample points. During this process, the LC block captures local signal features and adjusts sequence length by decoding inter-

sample relationships, setting the stage for subsequent global feature extraction. Our findings indicate that using different types of pooling layers for sequences of varying dimensions significantly impacts model performance. Specifically, employing an average pooling layer in the temporal branch and a max pooling layer in the frequency branch enables optimal model performance. We infer that incorporating information from all samples within the pooling range better represents the temporal sequence characteristics, while the maximum value of the samples within the pooling range more effectively captures signal fluctuation amplitudes, thereby better representing frequency sequence characteristics.

The value of DCCNNs lies in their ability to adjust the network's receptive field, addressing the "breadth" issue in signal decoding. Sequence splitting allows DCCNNs to train multiple signal decoding models, extracting features from each subsequence and thus enhancing the network's ability to capture global features in terms of "depth." High similarity between subsequences hinders the training of diverse feature extraction models in DCCNNs. Feature reconstruction of subsequences increases their diversity while maintaining continuity, thereby improving the model's efficiency in extracting global features. Sequence splitting and feature reconstruction effectively strengthen DCCNNs, and fine-tuning DCCNN hyperparameters according to sequence length significantly improves model performance.

In summary, the design of EEG-DBNet is effective, with each module significantly enhancing model performance. Experiments on two public datasets achieved optimal results. However, the numerous hyperparameters in EEG-DBNet make the tuning process cumbersome. Integrating the LC and GC blocks to reduce hyperparameters could improve the model's practicality. Additionally, exploring methods for extracting spatial signal features to further increase signal decoding dimensions is a key direction for future research.

References

- [1] P. Lahane, J. Jagtap, A. Inamdar, N. Karne, R. Dev, A review of recent trends in EEG based brain-computer interface, in: 2019 International Conference on Computational Intelligence in Data Science (ICCIDS), IEEE, 2019, pp. 1–6. doi:<https://doi.org/10.1109/ICCIDS.2019.8862054>.
- [2] E. Pirondini, M. Coscia, J. Minguillon, J. d. R. Millán, D. Van De Ville, S. Micera, EEG topographies provide subject-specific correlates of motor control, Scientific reports 7 (2017) 13229. doi:<https://doi.org/10.1038/s41598-017-13482-1>.
- [3] M. Orban, M. Elsamanty, K. Guo, S. Zhang, H. Yang, A review of brain activity and EEG-based brain-computer interfaces for rehabilitation application, Bioengineering 9 (2022) 768. doi:<https://doi.org/10.3390/bioengineering9120768>.
- [4] S. Natheir, S. Christie, R. Yilmaz, A. Winkler-Schwartz, K. Bajunaid, A. J. Sabbagh, P. Werthner, J. Fares, H. Azarnoush, R. Del Maestro, Utilizing artificial intelligence and electroencephalography to assess expertise on a simulated neurosurgical task, Computers in Biology and Medicine 152 (2023) 106286. doi:<https://doi.org/10.1016/j.combiomed.2022.106286>.
- [5] X. Zhao, H. Zhang, G. Zhu, F. You, S. Kuang, L. Sun, A multi-branch 3D convolutional neural network for EEG-based motor imagery classification, IEEE transactions on neural systems and rehabilitation engineering 27 (2019) 2164–2177. doi:<https://doi.org/10.1109/TNSRE.2019.2938295>.

- [6] L. Fraiwan, K. Lweesy, N. Khasawneh, H. Wenz, H. Dickhaus, Automated sleep stage identification system based on time–frequency analysis of a single EEG channel and random forest classifier, *Computer methods and programs in biomedicine* 108 (2012) 10–19. doi:<https://doi.org/10.1016/j.cmpb.2011.11.005>.
- [7] A. Subasi, M. I. Gursay, EEG signal classification using PCA, ICA, LDA and support vector machines, *Expert systems with applications* 37 (2010) 8659–8666. doi:<https://doi.org/10.1016/j.eswa.2010.06.065>.
- [8] J. Machado, A. Balbinot, Executed movement using EEG signals through a naive Bayes classifier, *Micromachines* 5 (2014) 1082–1105. doi:<https://doi.org/10.3390/mi5041082>.
- [9] M. Sha’Abani, N. Fuad, N. Jamal, M. Ismail, KNN and SVM classification for EEG: a review, in: *INECCE2019: Proceedings of the 5th International Conference on Electrical, Control & Computer Engineering*, Kuantan, Pahang, Malaysia, 29th July 2019, Springer, 2020, pp. 555–565. doi:https://doi.org/10.1007/978-981-15-2317-5_47.
- [10] 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. doi:<https://doi.org/10.3389/fnins.2012.00039>.
- [11] S. K. Bashar, A. R. Hassan, M. I. H. Bhuiyan, Motor imagery movements classification using multivariate EMD and short time fourier transform, in: *2015 Annual IEEE India Conference (INDICON)*, IEEE, 2015, pp. 1–6. doi:<https://doi.org/10.1109/INDICON.2015.7443813>.
- [12] A. Jafarifarmand, M. A. Badamchizadeh, EEG artifacts handling in a real practical brain–computer interface controlled vehicle, *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 27 (2019) 1200–1208. doi:<https://doi.org/10.1109/TNSRE.2019.2915801>.
- [13] K. Zhu, S. Wang, D. Zheng, M. Dai, Study on the effect of different electrode channel combinations of motor imagery EEG signals on classification accuracy, *The Journal of Engineering* 2019 (2019) 8641–8645. doi:<https://doi.org/10.1049/joe.2018.9073>.
- [14] R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer, M. Glasstetter, 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 (2017) 5391–5420. doi:<https://doi.org/10.1002/hbm.23730>.
- [15] 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. doi:<https://doi.org/10.1109/IJCNN.2008.4634130>.
- [16] 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 (2018) 056013. doi:<https://doi.org/10.1088/1741-2552/aace8c>.
- [17] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, Mobilenets: Efficient convolutional neural networks for mobile vision applications, *arXiv preprint arXiv:1704.04861* (2017). doi:<https://doi.org/10.48550/arXiv.1704.04861>.
- [18] X. Chen, X. Teng, H. Chen, Y. Pan, P. Geyer, Toward reliable signals decoding for electroencephalogram: A benchmark study to EEG-NeX, *Biomedical Signal Processing and Control* 87 (2024) 105475. doi:<https://doi.org/10.1016/j.bspc.2023.105475>.
- [19] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, K. Kavukcuoglu, Wavenet: A generative model for raw audio, *arXiv preprint arXiv:1609.03499* (2016). doi:<https://doi.org/10.48550/arXiv.1609.03499>.
- [20] T. M. Ingolfsson, M. Hersche, X. Wang, N. Kobayashi, L. Cavigelli, L. Benini, EEG-TCNet: An accurate temporal convolutional network for embedded motor-imagery brain–machine interfaces, in: *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, IEEE, 2020, pp. 2958–2965. doi:<https://doi.org/10.1109/SMC42975.2020.9283028>.
- [21] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, G. D. Hager, Temporal convolutional networks for action segmentation and detection, in: *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 156–165. doi:<https://doi.org/10.1109/CVPR.2017.113>.
- [22] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778. doi:<https://doi.org/10.1109/CVPR.2016.90>.
- [23] H. Altaheri, G. Muhammad, M. Alsulaiman, Physics-informed attention temporal convolutional network for EEG-based motor imagery classification, *IEEE transactions on industrial informatics* 19 (2022) 2249–2258. doi:<https://doi.org/10.1109/TII.2022.3197419>.
- [24] 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). doi:<https://doi.org/10.48550/arXiv.1706.03762>.
- [25] A. Salami, J. Andreu-Perez, H. Gillmeister, EEG-ITNet: An explainable inception temporal convolutional network for motor imagery classification, *IEEE Access* 10 (2022) 36672–36685. doi:<https://doi.org/10.1109/ACCESS.2022.3161489>.
- [26] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9. doi:<https://doi.org/10.1109/CVPR.2015.7298594>.
- [27] G. A. Altuwaijri, G. Muhammad, H. Altaheri, M. Alsulaiman, A multi-branch convolutional neural network with squeeze-and-excitation attention blocks for EEG-based motor imagery signals classification, *Diagnostics* 12 (2022) 995. doi:<https://doi.org/10.3390/diagnostics12040995>.
- [28] 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. doi:<https://doi.org/10.1109/CVPR.2018.00745>.
- [29] 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). doi:<https://doi.org/10.1109/TNSRE.2023.3342331>.
- [30] R. Mane, E. Chew, K. Chua, K. K. Ang, N. Robinson, A. P. Vinod, S.-W. Lee, C. Guan, FBCNet: A multi-view convolutional neural network for brain–computer interface, *arXiv preprint arXiv:2104.01233* (2021). doi:<https://doi.org/10.48550/arXiv.2104.01233>.
- [31] 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. doi:<https://doi.org/10.1109/TNSRE.2021.3099908>.
- [32] H. Zhi, Z. Yu, T. Yu, Z. Gu, J. Yang, A multi-domain convolutional neural network for EEG-based motor imagery decoding, *IEEE Transactions on Neural Systems and Rehabilitation Engineering* (2023). doi:<https://doi.org/10.1109/TNSRE.2023.3323325>.
- [33] W.-Y. Hsu, Y.-W. Cheng, EEG-Channel-Temporal-Spectral-Attention correlation for motor imagery EEG classification, *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 31 (2023) 1659–1669. doi:<https://doi.org/10.1109/TNSRE.2023.3255233>.
- [34] C. Brunner, R. Leeb, G. Müller-Putz, A. Schlögl, G. Pfurtscheller, BCI Competition 2008–Graz data set A, *Institute for Knowledge Discovery (Laboratory of Brain-Computer Interfaces)*, Graz University of Technology 16 (2008) 1–6. doi:<https://doi.org/10.21227/katb-zv89>.
- [35] R. Leeb, C. Brunner, G. Müller-Putz, A. Schlögl, G. Pfurtscheller, BCI Competition 2008–Graz data set B, *Graz University of Technology, Austria* 16 (2008) 1–6.
- [36] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: *International conference on machine learning*, pmlr, 2015, pp. 448–456. doi:<https://doi.org/10.48550/arXiv.1502.03167>.
- [37] D.-A. Clevert, T. Unterthiner, S. Hochreiter, Fast and accurate deep network learning by exponential linear units (elus), *arXiv preprint arXiv:1511.07289* (2015). doi:<https://doi.org/10.48550/arXiv.1511.07289>.
- [38] C. Garbin, X. Zhu, O. Marques, Dropout vs. batch normalization: an empirical study of their impact to deep learning, *Multimedia tools and applications* 79 (2020) 12777–12815. doi:<https://doi.org/10.1007/s11042-019-08453-9>.
- [39] A. F. Agarap, Deep learning using rectified linear units (relu), *arXiv preprint arXiv:1803.08375* (2018). doi:<https://doi.org/10.48550/arXiv.1803.08375>.