

# CSP-Net: Common Spatial Pattern Empowered Neural Networks for EEG-Based Motor Imagery Classification

Xue Jiang<sup>a,b</sup>, Lubin Meng<sup>a,b</sup>, Xinru Chen<sup>a,b</sup>, Yifan Xu<sup>a,b</sup>, Dongrui Wu<sup>a,b,\*</sup>

<sup>a</sup>Key Laboratory of the Ministry of Education for Image Processing and Intelligent Control, School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China

<sup>b</sup>Shenzhen Huazhong University of Science and Technology Research Institute, Shenzhen 518063, China

## Abstract

Electroencephalogram-based motor imagery (MI) classification is an important paradigm of non-invasive brain-computer interfaces. Common spatial pattern (CSP), which exploits different energy distributions on the scalp while performing different MI tasks, is very popular in MI classification. Convolutional neural networks (CNNs) have also achieved great success, due to their powerful learning capabilities. This paper proposes two CSP-empowered neural networks (CSP-Nets), which integrate knowledge-driven CSP filters with data-driven CNNs to enhance the performance in MI classification. CSP-Net-1 directly adds a CSP layer before a CNN to improve the input discriminability. CSP-Net-2 replaces a convolutional layer in CNN with a CSP layer. The CSP layer parameters in both CSP-Nets are initialized with CSP filters designed from the training data. During training, they can either be kept fixed or optimized using gradient descent. Experiments on four public MI datasets demonstrated that the two CSP-Nets consistently improved over their CNN backbones, in both within-subject and cross-subject classifications. They are particularly useful when the number of training samples is very small. Our work demonstrates the advantage of integrating knowledge-driven traditional machine learning with data-driven deep learning in EEG-based brain-computer interfaces.

**Keywords:** Brain-computer interfaces, electroencephalogram, motor imagery, common spatial pattern, convolutional neural network

## 1. Introduction

A brain-computer interface (BCI) establishes a direct communication pathway that enables the human brain to interact with external devices [1]. Electroencephalogram (EEG), which records the electrical activities on the scalp of the brain, is the most widely used input signal in non-invasive BCIs due to its affordability and convenience [2]. EEG-based BCIs have been used in controlling robots [3], decoding speech [4], enhancing computer gaming experience [5], and so on.

Motor imagery (MI) [6] is a classical paradigm of EEG-based BCIs, where a subject imagines the movement of a body part, e.g., right hand, left hand, right foot, left foot, both feet, and/or tongue, without actually executing it. An MI induces changes in the sensory-motor rhythms (SMR) of corresponding areas of the cerebral cortex, which primarily involve modulations of the  $\mu$  rhythm (8-12Hz) and the  $\beta$  rhythm (14-30Hz) [7]. Specifically, when an MI starts, these rhythmic activities decrease, resulting in event-related desynchronization (ERD); at the end of an MI, these rhythmic activities increase, resulting in event-related synchronization (ERS) [8, 9]. Therefore, the detection of SMR patterns within specific areas of the cerebral cortex can be used to identify which body part the subject is imagining moving.

Many algorithms have been proposed for EEG-based MI classification. Common spatial pattern (CSP) [10, 11] is one of the most widely used and effective approaches, which converts the raw multi-channel EEG signals into more discriminative spatial patterns. It was initially proposed for binary classification, by designing spatial filters that maximize the variance ratio of the filtered signals of different classes [10]. Dornhege *et al.* [12] extended it to multi-class classification using a one-versus-the-rest strategy. Ang *et al.* [13] proposed filter bank CSP (FBCSP), which bandpass filters EEG signals into multiple frequency bands, extracts CSP features from each band, and then selects the most useful features for classification. Lotte *et al.* [14] introduced regularized CSP to enhance the robustness of CSP.

Recent years have witnessed significant increase in using deep learning for EEG signal decoding [15], which integrates feature extraction and classification into a single end-to-end network. Among various deep architectures, convolutional neural networks (CNNs) are the most prevalent for MI classification [16, 17]. For example, Schirrmeister *et al.* [18] proposed ShallowCNN and DeepCNN for raw EEG classification. ShallowCNN is inspired by FBCSP and includes components such as temporal convolution, spatial convolution, log-variance calculation and a classifier, each corresponding to a specific step in FBCSP. DeepCNN is similar but includes more convolutional and pooling layers. Lawhern *et al.* [19] introduced a compact EEGNet, which has demonstrated promising performance

\*Emails: xuejiang@hust.edu.cn (Xue Jiang), lubinmeng@hust.edu.cn (Lubin Meng), xrchen@hust.edu.cn (Xinru Chen), yfxu@hust.edu.cn (Yifan Xu), drwu@hust.edu.cn (Dongrui Wu). Dongrui Wu is the corresponding author.

across various BCI tasks, including MI classification. Inspired also by FBCSP, EEGNet uses a two-step sequence of temporal convolution followed by depthwise convolution. Recently, FBCNet [20] extends the FBCSP approach by utilizing a hierarchical architecture that enhances feature extraction through multi-dimensional filtering, allowing it to capture richer spatial and temporal patterns in EEG data. EEGConformer [21] adopts a transformer-like architecture, combining self-attention mechanisms with convolutional layers, which enables the model to learn long-range dependencies in EEG signals effectively.

Though these data-driven deep models have achieved promising performance in MI classification, they usually require a large amount of labeled training data, which may not be always available in practice. This highlights the need to incorporate prior knowledge into EEG networks, as it can help reduce the reliance on extensive labeled datasets. By integrating prior knowledge, models can leverage existing insights about EEG signal characteristics, enhancing their generalization capabilities and performance even in data-scarce environments. This paper proposes CSP empowered neural networks (CSP-Net), which more effectively integrate CSP and CNNs. More specifically, we propose two CSP-Nets, by embedding CSP into different layers of the CNN models. The first, CSP-Net-1, places a CSP layer before a CNN to filter the EEG signals for enhancing their discriminability. The second, CSP-Net-2, replaces a CNN's convolutional layer with a CSP layer to provide task-specific prior knowledge initialization. The parameters in the CSP layer of both CSP-Nets are initialized from CSP filters designed on the training data. They can be fixed or optimized by gradient descent during training. In summary, CSP-Nets integrate the strengths of traditional CSP feature extraction with deep learning by embedding CSP layers in CNN architectures. This approach enhances the model's ability to capture relevant features from EEG signals, making it a more effective solution for MI classification. Our main contributions are:

- Integration of CSP and CNNs: We propose a novel framework that combines CSP with CNNs for MI classification, enhancing EEG feature extraction and improving classification performance.
- Two CSP-Net Variants: We introduce two architectures, CSP-Net-1, which incorporates a CSP layer before the CNN, and CSP-Net-2, which replaces a convolutional layer with a CSP layer for task-specific prior knowledge initialization. Both variants allow CSP layer parameters to be fixed or further optimized.
- Performance on Multiple EEG Datasets: CSP-Nets demonstrate strong performance across various scenarios, including within-subject and cross-subject classifications, as well as in small sample settings. The models demonstrate generalization across different backbone architectures, validated on four public MI datasets.

The rest of this paper is structured as follows. Section 2 introduces the classical CSP and proposes two CSP-Nets. Section 3 presents the experimental settings and experimental results. Finally, Section 4 draws conclusions.

## 2. Methods

This section introduces the CSP algorithm, five CNN models for MI classification, and our proposed two CSP-Nets to integrate CSP and CNNs.

### 2.1. CSP

CSP was first proposed by Koles *et al.* [22] to extract discriminative features from EEG signals of two human populations. Müller-Gerking *et al.* [23] later extended it to MI classification. Since then, it has become one of the most popular and effective algorithms in MI-based BCIs [10, 11].

Fig. 1 shows *t*-SNE visualization of some real EEG trials before and after CSP filtering from Subject 3 of Dataset 2a in BCI Competition IV [24]. Clearly, after CSP filtering, samples from different classes become more distinguishable.

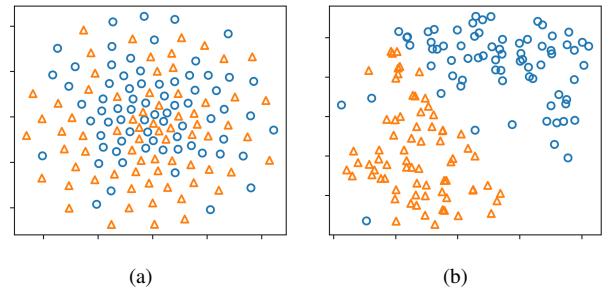


Figure 1: *t*-SNE visualization of (a) the raw EEG trials; and, (b) the CSP-filtered trials. Different shapes (colors) represent different classes.

For binary classification, CSP aims to learn spatial filters that maximize the variance of EEG signals from one class while simultaneously minimizing the variance from the other class. Let  $X_i \in \mathbb{R}^{c \times t}$  be an EEG trial of MI task  $i$ , where  $i \in \{1, 2\}$  is the class index,  $c$  the number of channels, and  $t$  the number of time domain samples. CSP generates a spatial filtering matrix  $W \in \mathbb{R}^{c \times f}$  ( $f < c$ ) that projects the original EEG trials into a lower-dimensional space with higher discriminability.  $W$  is obtained by maximizing (or minimizing):

$$J(W) = \frac{W^\top \bar{X}_1 \bar{X}_1^\top W}{W^\top \bar{X}_2 \bar{X}_2^\top W} = \frac{W^\top \bar{C}_1 W}{W^\top \bar{C}_2 W}, \quad (1)$$

where  $\bar{X}_i \in \mathbb{R}^{c \times t}$  is the averaged EEG trial from class  $i$ , and  $\bar{C}_i \in \mathbb{R}^{c \times c}$  the mean spatial covariance matrix of all EEG trials in class  $i$ .

Since  $J(W) = J(kW)$  for any arbitrary real constant  $k$ , maximizing  $J(W)$  is equivalent to maximizing  $W^\top \bar{C}_1 W$ , subject to the constraint  $W^\top \bar{C}_2 W = I_f$ . This optimization problem can be solved using the Lagrange multiplier method [14], whose Lagrange function is

$$F(W, \lambda) = W^\top \bar{C}_1 W - \lambda(W^\top \bar{C}_2 W - I_f). \quad (2)$$

Setting the derivative of  $F(W, \lambda)$  with respect to  $W$  to 0, we

have

$$\begin{aligned}\frac{\partial F(W, \lambda)}{\partial W} &= 2W^\top \bar{C}_1 - 2\lambda W^\top \bar{C}_2 = 0 \\ \Leftrightarrow \bar{C}_1 W &= \lambda \bar{C}_2 W \\ \Leftrightarrow \bar{C}_2^{-1} \bar{C}_1 W &= \lambda W,\end{aligned}$$

which becomes a standard eigenvalue decomposition problem.

The spatial filtering matrix  $W$  consists of eigenvectors corresponding to the  $\frac{f}{2}$  largest and the  $\frac{f}{2}$  smallest eigenvalues of  $\bar{C}_2^{-1} \bar{C}_1$ .

## 2.2. CNNs for MI Classification

Five popular CNN models are considered in this paper: EEGNet [19], DeepCNN [18], ShallowCNN [18], FBCNet [20], and EEGConformer [21]. Their architectures are detailed in Tables 1-5, respectively.

- EEGNet [19], which consists of three convolutional blocks and a classifier block. The first convolutional block performs temporal filtering for capturing frequency information. The second spatial filter block uses depthwise convolution with size  $(c, 1)$  to learn spatial filters. The third separable convolutional block is used to reduce the number of parameters and decouple the relationships within and across feature maps.
- DeepCNN [18], compared with EEGNet, it is deeper and hence has much more parameters. It mainly includes a temporal convolutional block, a spatial filter block, two standard convolutional blocks and a classifier block. The first temporal and spatial convolutional blocks are specially designed to handle EEG inputs and the other two are standard ones.
- ShallowCNN [18], which is a shallow version of DeepCNN, inspired by FBCSP. Its first two blocks are similar to the temporal and spatial convolutional blocks of DeepCNN, but with a larger kernel, a different activation function, and a different pooling approach.
- FBCNet [20], which is a simple yet effective CNN architecture. It begins by applying multiple fixed-parameter band-pass filters to decompose the EEG into various frequency bands as multi-view inputs. Spatial filter block is then used to extract spatially discriminative patterns from each frequency band. Finally, a classifier block is designed for classification.
- EEGConformer [21], which is a compact convolutional transformer model. The convolution module also includes a temporal convolutional block and a spatial filter block for learning the low-level local features. The multiple self-attention modules are used to extract the global correlation within the local features.

Table 1: EEGNet [19].

Block	Layer	Filter size	Number of filters
Temporal convolution	Conv2D	$(1, \frac{f_s}{2})$	4
	Batch normalization	-	-
Depthwise spatial filter	DepthwiseConv2D	$(c, 1)$	8
	Batch normalization	-	-
	ELU activation	-	-
	Average pooling	$(1, 4)$	-
	Dropout	-	-
Separable convolution	SeparableConv2D	$(1, 16)$	8
	Batch normalization	-	-
	PointwiseConv2D	$(1, 1)$	8
	Batch normalization	-	-
	ELU activation	-	-
	Average pooling	$(1, 8)$	-
Classifier	Dropout	-	-
	Fully connection	-	-

Table 2: DeepCNN [18].

Block	Layer	Filter size	Number of filters
Temporal convolution	Conv2D	$(1, 5)$	25
Spatial filter	Conv2D	$(c, 1)$	25
	Batch normalization	-	-
	ELU activation	-	-
	Max pooling	$(1, 2)$	-
	Dropout	-	-
Standard convolution	Conv2D	$(1, 5)$	50
	Batch normalization	-	-
	ELU activation	-	-
	Max pooling	$(1, 2)$	-
	Dropout	-	-
Standard convolution	Conv2D	$(1, 5)$	100
	Batch normalization	-	-
	ELU activation	-	-
	Max pooling	$(1, 2)$	-
	Dropout	-	-
Classifier	Fully connection	-	-

Table 3: ShallowCNN [18].

Block	Layer	Filter size	Number of filters
Temporal convolution	Conv2D	$(1, 13)$	40
Spatial filter	Conv2D	$(c, 1)$	40
	Batch Normalization	-	-
	Squaring Activation	-	-
	Average Pooling	$(1, 35)$	-
	Logarithmic Activation	-	-
Classifier	Dropout	-	-
	Fully Connection	-	-

Table 4: FBCNet [20].

Block	Layer	Filter size	Number of filters
Band-pass filter	-	-	6
Spatial filter	DepthwiseConv2D	$(c, 1)$	48
	Batch normalization	-	-
	Swish activation	-	-
	Variance layer	-	-
Classifier	Fully connection	-	-

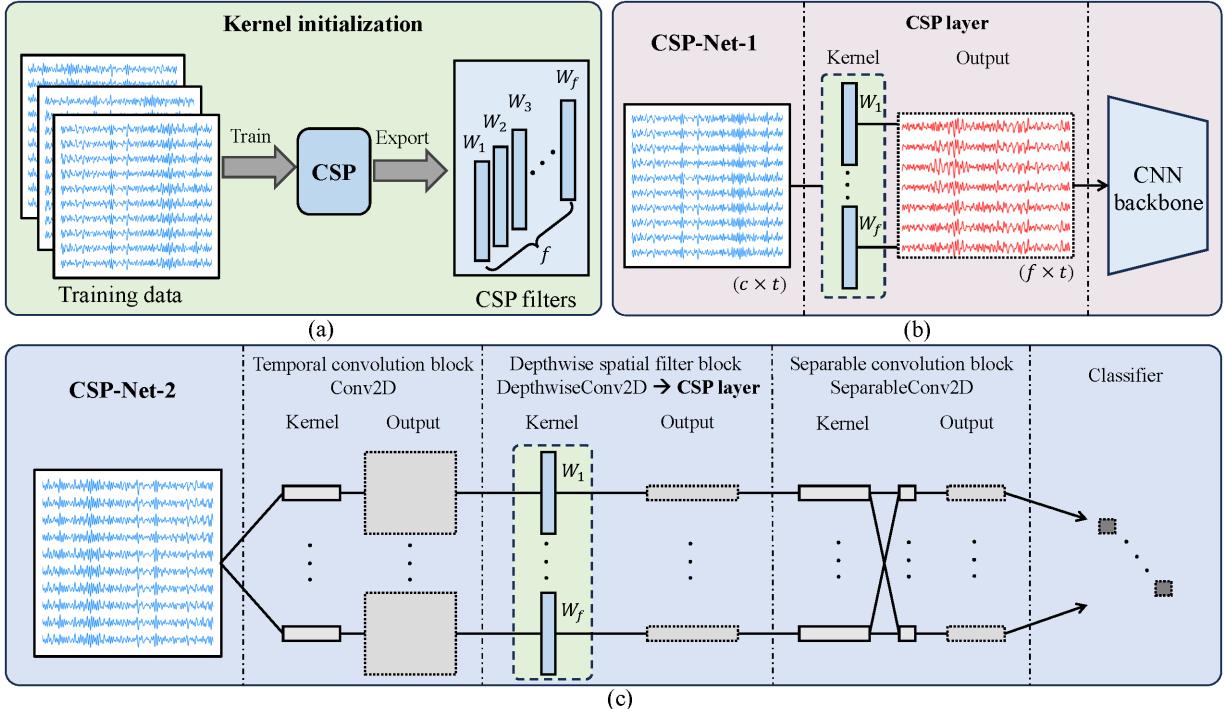


Figure 2: Our proposed CSP-Nets. (a) Traditional CSP filters are used to initialize the CSP layer in CSP-Nets. (b) CSP-Net-1, which directly adds a CSP layer before a CNN backbone. (c) CSP-Net-2, illustrated using EEGNet [19] (Table 1 in Supplementary Materials); the DepthwiseConv2D layer in its depthwise spatial filter block is replaced by a CSP layer.

Table 5: EEGConformer [21].

Block	Layer	Filter size	Number of filters
Temporal convolution	Conv2D	(1, 25)	40
Spatial filter	Conv2D	(c, 1)	40
	Batch normalization	-	-
	ELU activation	-	-
	Average pooling	(1, 75)	-
	Dropout	-	-
	PointwiseConv2D	(1, 1)	40
6× Self-attention	Rearrange	-	-
	Layer normalization	-	-
	MHA	-	-
	Dropout	-	-
	Residual add	-	-
	Layer normalization	-	-
	FFN	-	-
	Dropout	-	-
	Residual add	-	-

### 2.3. CSP-Net-1

Our proposed CSP-Net-1 simply performs CSP before a CNN.

As illustrated in Fig. 2(a), all training EEG samples are used in CSP, resulting in  $f$  spatial filters  $W_i \in \mathbb{R}^{c \times 1}$ ,  $i = 1, \dots, f$ . Then, as shown in Fig. 2(b), CSP-Net-1 uses these filters to spatially filter the raw EEG signals, before passing them to a CNN backbone.

There could be two different training approaches: 1) Fix the CSP layer and train the CNN backbone only (CSP-Net-1-fix); and, 2) update the CSP layer and the CNN backbone simultaneously (CSP-Net-1-upd). Their effectiveness will be discussed in Section 3.3.

CSP-Net-1 applies CSP filtering as a pre-processing step, enabling the model to work with more discriminative input signals. This explicit inclusion of the CSP filter provides a more structured way to embed expert knowledge into the network, thereby improving the model’s capacity to capture task-relevant spatial features.

Algorithm 1 gives the pseudo-code of CSP-Net-1.

---

#### Algorithm 1: CSP-Net-1 for MI classification.

---

**Input:** Training data  $X$ ; a CNN model.

- 1 Perform CSP on  $X$  to obtain the filter matrix  $W$ ;
  - 2 Initialize CSP-Net-1, which consists of a CSP layer with weights  $W$  and a randomly initialized CNN;
  - 3 Train CSP-Net-1 on  $X$ ;
  - 4 **return** CSP-Net-1.
- 

### 2.4. CSP-Net-2

Many CNN models have been proposed for MI classification, which typically consist of multiple convolution-pooling layers for feature extraction and some fully connected layers for classification. Although they differ in architecture, they usually include a spatial filter layer with spatial convolutional kernels specifically designed for EEG signals.

CSP-Net-2 replaces their spatial filter layer with a CSP layer. Fig. 2(c) uses EEGNet as the CNN backbone to illustrate the architecture of CSP-Net-2. For clarity, we primarily depict the

---

**Algorithm 2:** CSP-Net-2 for MI classification.

---

**Input:** Training data  $X$ ; a CNN model.

- 1 Perform CSP on  $X$  to obtain the filter matrix  $W$ ;
  - 2 Randomly initialize the CNN model;
  - 3 Initialize CSP-Net-2, by replacing the convolutional layer in the spatial filter block of the CNN model by a CSP layer with weights  $W$ ;
  - 4 Train CSP-Net-2 on  $X$ ;
  - 5 **return** CSP-Net-2.
- 

connection of the convolutional kernel between inputs and outputs. The depthwise spatial filter block aims to learn spatial patterns in EEG data. CSP-Net-2 replaces the convolutional kernels in this block with the CSP filters, and keeps other parts unchanged.

More specifically, CSP-Net-2 uses the CSP layer to replace the DepthwiseConv2D layer in the spatial filter block of EEG-Net (8 kernels), the Conv2D layer in the spatial filter block of DeepCNN (25 kernels), the Conv2D layer in the spatial filter block of ShallowCNN (40 kernels), the DepthwiseConv2D layer in the spatial filter block of FBCNet (48 kernels), and the Conv2D layer in the spatial filter block of EEGConformer (40 kernels).

Similar to CSP-Net-1, the CSP filter layer in CSP-Net-2 can either be fixed (CSP-Net-2-fix) or updated (CSP-Net-2-upd). Furthermore, this replacement is significant as it allows CSP-Net-2 to explicitly incorporate prior knowledge about spatial filtering, enhancing the model’s ability to capture discriminative features from the EEG signals. The flexibility of using either fixed or updated CSP filters also provides a balance between stability and adaptability during training, which we discussed in detail in Section 3.3.

Algorithm 2 gives the pseudo-code of CSP-Net-2.

### 3. Experiments and Results

This section presents the experimental results to validate the effectiveness of our proposed CSP-Nets.

#### 3.1. Datasets

Four public MI datasets from BNCI-Horizon<sup>1</sup>, summarized in Table 6, were used in our experiments:

1. MI4C and MI2C: They were from the 001-2014 dataset. The EEG signals were sampled at 250Hz. MI2C includes only left-hand and right-hand trials. MI4C includes all classes.
2. MI14S: This was from the 002-2014 dataset. The EEG signals were sampled at 512Hz.
3. MI9S: This was the 001-2015 dataset. The EEG signals were recorded at 512Hz. The last three subjects were discarded due to their poor performance [25, 26].

They were downloaded and pre-processed using the MOABB framework [27]. All datasets were pre-processed with an 8-32Hz bandpass filter.

Table 6: Summary of the four MI datasets.

Datasets	# Subjects	# Channels	# Trials per subject	# Classes
MI4C	9	22	288	4
MI2C	9	22	144	2
MI14S	14	15	100	2
MI9S	9	13	200	2

#### 3.2. Implementation Details

We evaluated the performance of CSP-Nets in both within-subject and cross-subject classifications:

1. Within-subject classification: For each individual subject, 80% trials were used for training, and the remaining 20% for testing.
2. Cross-subject classification: Leave-one-subject-out cross-validation was performed, i.e., one subject was used as the test set and all remaining ones as the training set.

All experiments were repeated 5 times, and the average accuracies are reported.

We used Adam optimizer with batch size 128 and initial learning rate 0.01, and cross-entropy loss with weight decay 0.0005. The maximum number of training epochs was 200. The CSP layer used by default  $f = 8$  spatial filters (Section 3.6 presents sensitivity analysis). For CSP-Net-2, the number of convolutional kernels in the original spatial filter layer of the CNN models may be larger than 8. We expanded the CSP filters to address this mismatch: when the number of convolutional kernels exceeds the number of CSP filters, we replicate the CSP filters to match the number of required kernels. Specifically, we duplicated the 8 CSP filters 5 times to match the 40 kernels in the spatial filter block of ShallowCNN and EEGConformer ( $8 \times 5 = 40$ ), duplicated the 8 CSP filters 6 times to match the 48 kernels in the spatial filter block of FBCNet ( $8 \times 6 = 48$ ), and duplicated the 8 CSP filters 3 times and randomly selected one more to match the 25 kernels in the spatial filter block of DeepCNN ( $8 \times 3 + 1 = 25$ ).

#### 3.3. Experimental Results

Table 7 shows the classification accuracies for the individual subjects on MI4C, where CSP-LR used logistic regression as the classifier. Tables 8-10 show the average classification results across all subjects on the other three datasets, due to page limit. We performed paired  $t$ -tests on the results, calculated  $p$ -values between the standard backbone models and the CSP-Nets, and adjusted them using Benjamini Hochberg False Discovery Rate correction. Observe that:

1. Both CSP-Nets were generally highly effective on all datasets and backbones. Embedding CSP knowledge in CNN backbones resulted in significant performance improvements. For example, in within-subject classification

<sup>1</sup><http://www.bnci-horizon-2020.eu/database/data-sets>

on MI4C, CSP-Net-2-fix increased the average accuracy on all subjects from 63.50% to 71.91% after integrating CSP information into EEGNet as CSP-Net-2-fix. The average accuracies across all five backbones also exhibited significant improvements, from an initial accuracy of 61.32% to as high as 67.33%.

2. CSP-Nets with fixed CSP layer parameters generally performed better. Particularly, CSP-Net-2-fix achieved substantial improvements over EEGNet, DeepCNN, and EEGConformer. This validated that the incorporation of CSP prior knowledge can enhance the generalization of CNN models. High number and proportion of parameters of spatial convolutional kernels in ShallowCNN and FBC-Net may overshadow the benefits offered by CSP filters.
3. CSP-Nets had larger performance improvements in within-subject classification than cross-subject classification. This might be because: 1) within-subject classification had much fewer training samples than cross-subject classification, and hence prior knowledge in CSP is more helpful to the generalization performance; and, 2) cross-subject classification is intrinsically more challenging, as there are large individual differences among different subjects. The impact of training data quantity on CSP-Nets is discussed in Section 3.5.
4. CSP-Nets achieved better performance on most subjects. However, for some subjects where CSP did not perform well, CSP-Nets also struggled, e.g., Subject 1, 2, 5 and 6 in cross-subject classification on MI4C.

### 3.4. Comparative Performance Analysis

We further compared our approaches with nine other approaches, including the state-of-the-art traditional approaches and deep learning approaches: CSP [10], FBCSP [13], MDRM [28], DeepCNN [18], LMDA-Net [29], ShallowCNN [18], EEGConformer [21], EEGNet [19], and FBC-Net [20]. Table 11 presents the classification accuracies of CSP-Net-1 and CSP-Net-2 compared to these baselines. In CSP-Net-1 and CSP-Net-2, the EEGNet was used as the backbone architecture, and the fixed CSP layer was applied. The same training and test data were used for all models.

Both CSP-Net-1 and CSP-Net-2 demonstrated superior performance compared to traditional approaches like CSP and FBCSP, as well as more recent models like FBCNet and EEG-Conformer. These results highlight their effectiveness for EEG signal classification tasks.

### 3.5. Small Sample Setting

Deep CNN models may easily overfit when the training dataset is small. Figs. 3-6 show the accuracy improvements compared with the backbone at different training data ratios (the number of training samples used to train the model divided by the total number training samples) on the four datasets, respectively.

Observe that:

1. Consistent with previous findings, CSP-Net-fix generally outperformed CSP-Net-upd, with CSP-Net-2-fix particularly competitive on EEGNet and DeepCNN. For example, in within-subject classification, CSP-Net-2-fix achieved a remarkable accuracy improvement of more than 20% over the DeepCNN backbone, when trained with only 50% of the training data on MI4C.
2. Overall, the performance improvements of CSP-Nets were more obvious when the training data size was small. This may be because the embedding of prior knowledge greatly reduces the overfitting issue of CNN backbones in small sample scenarios.

### 3.6. Influence of the Number of CSP Filters

We further investigated the influence of the number of CSP filters ( $f$ ) on the performance of CSP-Nets with three backbones on MI4C. The dataset includes 22-channel EEG signals, so we considered  $f \in \{4, 8, 12, 16, 22\}$ . Fig. 7 shows the corresponding accuracies of the two CSP-Nets (fixed CSP layer). As the number of filters increased, the accuracy first increased and then decreases, which is intuitive. Generally,  $f = 8$  seems to be a good choice to balance the performance and computational cost.

### 3.7. Ablation Studies

An ablation study was performed to verify that the performance improvement of CSP-Net-1 was not due to an increase in the number of network parameters.

Specifically, we trained CSP-Net-1-rad, which replaced the CSP layer of CSP-Net-1 with a randomly initialized layer of the same size. The results on the four MI datasets in within-subject classification are shown in Table 12. Generally, CSP-Net-1-rad performed similarly to the standard backbone, suggesting that the performance improvement of CSP-Net-1 was due to its incorporation of knowledge from CSP, instead of more parameters.

### 3.8. Training Process Visualization

Fig. 8 shows the average cross-subject training and test accuracy curves of CSP-Nets (fixed CSP layer) and their corresponding backbones (EEGNet) on the four datasets. For all the backbones, there was a large gap between the training and test curves, indicating overfitting. Our proposed CSP-Nets effectively leveraged the knowledge from the CSP filters for better initialization, reducing the gap and achieving better test performance.

### 3.9. Visualization of the CSP Filters

We further visualized the spatial convolutional kernel weights from the CSP-Net-2 and the counterparts from standard backbone. In Fig. 9, we present the eight spatial filters in the EEGNet model and the CSP filters in the CSP-Net-2-fix model for within-subject classification on Subject 1 of MI2C (binary classification on the left hand and right hand). We can observe that the CSP filters in CSP-Net-2-fix exhibited a more focused and obvious left-right distribution concentrated on a

Table 7: Classification accuracies (%) on MI4C. Average accuracies higher than Standard are marked in bold. Asterisks indicate statistically significant differences between standard backbone and CSP-Net under adjusted paired *t*-test, where \* means  $p < 0.05$ , \*\* means  $p < 0.01$ , \*\*\* means  $p < 0.001$ .

Scenario	Backbone	Approach	S1	S2	S3	S4	S5	S6	S7	S8	S9	Average acc±std	
Within-Subject	EEGNet	-	CSP-LR	71.85	61.27	78.64	48.52	35.32	40.14	69.07	75.00	70.37	61.13
		Standard	72.59	49.35	81.36	44.51	49.31	39.29	65.98	84.19	84.93	63.50	
		CSP-Net-1-upd	83.40	57.31	88.26	50.36	58.36	43.90	75.52	84.01	87.59	69.86***±1.54	
		CSP-Net-1-fix	81.43	58.73	90.70	53.48	53.73	45.74	82.42	83.95	86.88	70.79***±1.68	
		CSP-Net-2-upd	77.83	56.85	86.91	47.95	46.24	41.35	74.20	80.74	84.11	66.24±2.89	
		CSP-Net-2-fix	81.18	63.55	92.36	52.69	56.75	46.27	83.02	82.64	88.76	71.91***±0.74	
	DeepCNN	Standard	56.86	45.93	66.55	39.72	25.48	30.14	57.34	68.66	72.93	51.51±1.13	
		CSP-Net-1-upd	69.07	57.07	80.32	50.76	38.34	37.82	68.72	73.76	78.98	61.65***±1.70	
		CSP-Net-1-fix	70.12	56.24	81.90	52.06	46.05	38.28	69.04	72.94	82.85	63.28***±1.26	
		CSP-Net-2-upd	62.52	41.36	59.02	43.28	25.64	25.00	54.47	70.01	71.74	50.34±0.95	
		CSP-Net-2-fix	73.07	55.34	82.59	51.55	45.90	38.18	74.78	76.61	81.52	64.39***±2.46	
	ShallowCNN	Standard	69.35	57.51	74.62	53.04	50.74	46.14	76.48	76.33	81.95	65.13±1.38	
		CSP-Net-1-upd	79.99	57.93	84.78	57.87	54.98	44.48	85.43	80.79	80.92	69.69***±1.58	
		CSP-Net-1-fix	80.15	59.93	86.51	58.04	52.83	44.48	85.14	83.04	83.64	70.42***±1.41	
		CSP-Net-2-upd	70.21	57.53	79.94	50.79	37.88	40.42	76.08	83.20	80.67	64.08±1.62	
		CSP-Net-2-fix	69.04	63.02	86.55	45.62	31.48	38.89	77.06	79.06	82.27	63.66±2.01	
Cross-Subject	FBCNet	Standard	69.27	53.87	83.77	50.17	53.03	43.78	69.61	80.22	83.95	65.30±1.76	
		CSP-Net-1-upd	77.39	53.70	86.65	52.61	60.79	45.62	78.23	86.81	84.71	69.61***±1.23	
		CSP-Net-1-fix	76.41	56.87	87.44	49.11	53.56	42.79	77.72	86.07	86.01	68.44**±2.04	
		CSP-Net-2-upd	71.19	54.88	87.29	53.58	54.56	43.96	72.34	83.05	86.46	67.48*±1.85	
		CSP-Net-2-fix	76.74	57.42	86.45	46.79	53.81	43.64	71.47	82.65	85.98	67.22±1.02	
	EEGConformer	Standard	75.00	46.97	79.81	50.23	37.46	46.05	72.25	77.19	65.42	61.15±1.67	
		CSP-Net-1-upd	81.98	63.81	87.68	58.57	60.62	52.46	88.04	82.96	71.73	71.98***±1.38	
		CSP-Net-1-fix	84.28	59.84	87.41	61.85	58.49	53.90	90.18	79.96	71.82	71.97***±1.14	
		CSP-Net-2-upd	80.49	58.97	89.79	60.50	56.39	55.85	84.24	85.14	75.04	71.82***±1.04	
		CSP-Net-2-fix	82.55	60.73	86.94	45.70	58.18	46.37	78.67	83.13	82.95	69.47***±1.04	
	Average	Standard	68.61	50.73	77.22	47.53	43.20	41.08	68.33	77.32	77.84	61.32±1.57	
		CSP-Net-1-upd	<b>78.37</b>	<b>57.96</b>	<b>85.54</b>	<b>54.03</b>	<b>54.62</b>	<b>44.86</b>	<b>79.19</b>	<b>81.67</b>	<b>80.79</b>	<b>68.56***±1.49</b>	
		CSP-Net-1-fix	<b>78.48</b>	<b>58.32</b>	<b>86.79</b>	<b>54.91</b>	<b>52.93</b>	<b>45.04</b>	<b>80.90</b>	<b>81.19</b>	<b>82.24</b>	<b>68.98***±1.51</b>	
		CSP-Net-2-upd	<b>72.45</b>	<b>53.92</b>	<b>80.59</b>	<b>51.22</b>	<b>44.14</b>	<b>41.32</b>	<b>72.27</b>	<b>80.43</b>	<b>79.60</b>	<b>63.99***±1.67</b>	
		CSP-Net-2-fix	<b>76.52</b>	<b>60.01</b>	<b>86.98</b>	<b>48.47</b>	<b>49.22</b>	<b>42.67</b>	<b>77.00</b>	<b>80.82</b>	<b>84.30</b>	<b>67.33***±1.45</b>	
Cross-Subject	FBCNet	-	CSP-LR	61.46	22.92	72.22	43.06	32.29	39.58	62.50	76.04	62.85	52.55
		Standard	68.96	30.35	71.88	38.06	37.01	36.74	42.01	58.19	61.74	49.44±1.75	
		CSP-Net-1-upd	65.62	32.15	72.92	41.39	34.93	38.68	55.28	65.69	61.74	52.04*±1.48	
		CSP-Net-1-fix	62.50	32.43	76.32	40.97	34.58	37.43	52.01	67.64	65.62	52.17*±1.13	
		CSP-Net-2-upd	66.94	32.64	74.17	42.01	35.07	40.97	42.78	67.08	63.82	51.72*±1.59	
		CSP-Net-2-fix	66.46	31.53	74.44	39.65	31.04	37.29	53.40	63.06	68.61	51.72±1.38	
	DeepCNN	Standard	65.35	34.03	52.99	37.92	38.19	43.40	41.81	59.24	53.47	47.38±2.10	
		CSP-Net-1-upd	61.32	33.40	67.57	41.81	35.56	40.90	43.19	63.19	57.01	49.33±0.70	
		CSP-Net-1-fix	60.07	34.79	62.22	42.50	35.21	40.97	41.67	67.01	60.14	49.40±0.53	
		CSP-Net-2-upd	64.93	31.18	65.00	38.96	39.58	41.67	48.61	56.94	57.15	49.34±1.11	
		CSP-Net-2-fix	62.01	29.93	69.44	37.99	36.32	39.44	52.92	63.33	61.67	50.34*±1.81	
	ShallowCNN	Standard	68.12	33.40	71.04	41.46	36.18	45.49	43.82	69.51	64.17	52.58±0.65	
		CSP-Net-1-upd	66.25	30.63	70.69	41.53	32.15	40.49	50.00	64.65	59.03	50.60±1.23	
		CSP-Net-1-fix	68.12	31.04	72.01	42.71	32.78	39.24	53.96	69.38	61.60	52.31±1.28	
		CSP-Net-2-upd	66.39	30.63	72.57	43.12	32.29	43.06	47.29	67.57	63.61	51.84±1.25	
		CSP-Net-2-fix	62.15	28.47	72.08	39.58	32.36	41.67	54.93	67.64	67.64	51.84±1.04	
Subject	FBCNet	Standard	62.92	31.39	62.22	41.67	31.25	36.25	40.76	55.21	57.01	46.52±1.49	
		CSP-Net-1-upd	66.18	30.90	62.50	42.43	31.53	38.33	43.26	57.43	60.49	48.12±1.31	
		CSP-Net-1-fix	67.22	31.39	67.43	39.51	31.11	38.06	45.90	60.76	60.28	49.07***±1.11	
		CSP-Net-2-upd	64.65	30.42	61.94	41.39	30.83	37.92	40.69	56.88	57.64	46.93±1.27	
		CSP-Net-2-fix	57.50	32.50	64.38	40.35	36.67	36.88	49.79	64.31	65.00	49.71**±1.24	
	EEGConformer	Standard	52.64	27.64	49.51	33.68	32.29	39.86	30.14	54.51	41.60	40.21±1.67	
		CSP-Net-1-upd	61.94	35.07	63.06	34.38	32.36	37.85	26.11	61.39	55.76	45.32***±0.90	
		CSP-Net-1-fix	58.54	35.83	63.33	35.35	32.64	38.19	27.78	64.03	56.18	45.76***±1.09	
		CSP-Net-2-upd	55.76	33.06	71.67	38.19	33.19	35.14	44.03	61.39	63.75	48.46***±0.62	
		CSP-Net-2-fix	56.81	33.26	72.92	39.65	31.11	37.57	44.44	67.22	65.83	49.87***±0.94	
	Average	Standard	63.60	31.36	61.53	38.56	34.98	40.35	39.71	59.33	55.60	47.22±1.53	
		CSP-Net-1-upd	<b>64.26</b>	<b>32.43</b>	<b>67.35</b>	<b>40.31</b>	33.31	39.25	<b>43.57</b>	<b>62.47</b>	<b>58.81</b>	<b>49.08***±1.12</b>	
		CSP-Net-1-fix	63.29	<b>33.10</b>	<b>68.26</b>	<b>40.21</b>	33.26	38.78	<b>44.26</b>	<b>65.76</b>	<b>60.76</b>	<b>49.74***±1.03</b>	
		CSP-Net-2-upd	63.73	31.59	<b>69.07</b>	<b>40.73</b>	34.19	39.75	<b>44.68</b>	<b>61.97</b>	<b>61.19</b>	<b>49.66***±1.17</b>	
		CSP-Net-2-fix	60.99	31.14	<b>70.65</b>	<b>39.44</b>	33.50	38.57	<b>51.10</b>	<b>65.11</b>	<b>65.75</b>	<b>50.69***±1.28</b>	

Table 8: Average classification accuracies (%) on MI2C. Those higher than Standard are marked in bold. Asterisks indicate statistically significant differences between standard backbone and CSP-Net under adjusted paired *t*-test, where \* means  $p < 0.05$ , \*\* means  $p < 0.01$ , \*\*\* means  $p < 0.001$ .

Scenario	Approach	Backbone					
		EEGNet	DeepCNN	ShallowCNN	FBCNet	EEGConformer	Average acc±std
Within-subject	CSP-LR	-	-	-	-	-	75.72
	Standard	76.38±2.21	61.46±3.28	76.29±2.92	78.11±2.64	75.31±1.45	73.51±2.50
	CSP-Net-1-upd	<b>80.02</b> ±2.80	<b>70.86</b> ***±3.58	<b>82.50</b> **±2.60	<b>80.70</b> *±1.85	<b>81.06</b> ***±1.46	<b>79.02</b> ***±2.45
	CSP-Net-1-fix	<b>81.69</b> *±0.49	<b>70.37</b> ***±3.41	<b>83.71</b> ***±1.45	<b>82.39</b> **±2.70	<b>82.05</b> ***±0.88	<b>80.04</b> ***±1.78
	CSP-Net-2-upd	75.94±2.24	<b>61.59</b> ±1.93	<b>77.33</b> ±1.68	<b>79.65</b> ±2.48	<b>81.53</b> ***±2.80	<b>75.21</b> **±2.23
	CSP-Net-2-fix	<b>79.66</b> *±2.38	<b>75.86</b> ***±1.11	75.34±1.81	<b>79.54</b> ±0.58	<b>81.18</b> ***±1.61	<b>78.32</b> ***±1.50
Cross-subject	CSP-LR	-	-	-	-	-	72.92
	Standard	<b>71.50</b> ±0.87	73.15±1.37	74.32±1.50	69.34±1.50	65.25±1.03	70.71±1.16
	CSP-Net-1-upd	<b>73.53</b> *±1.39	<b>74.86</b> *±0.73	<b>74.65</b> ±0.63	<b>71.00</b> ±1.60	<b>70.94</b> ***±0.67	<b>73.00</b> ***±1.00
	CSP-Net-1-fix	<b>74.51</b> ***±1.28	<b>74.75</b> ±1.36	<b>75.51</b> ±0.65	<b>73.09</b> ***±1.35	<b>71.20</b> ***±0.57	<b>73.81</b> ***±1.04
	CSP-Net-2-upd	<b>72.31</b> ±1.23	72.93±0.48	70.40±0.87	<b>69.41</b> ±1.71	<b>70.76</b> ***±1.37	<b>71.16</b> ±1.13
	CSP-Net-2-fix	<b>75.25</b> ***±1.28	<b>73.38</b> ±0.91	<b>76.11</b> ±0.80	<b>73.61</b> ***±0.27	<b>72.95</b> ***±0.55	<b>74.26</b> ***±0.76

Table 9: Average classification accuracies (%) on MI14S. Those higher than Standard are marked in bold. Asterisks indicate statistically significant differences between standard backbone and CSP-Net under adjusted paired *t*-test, where \* means  $p < 0.05$ , \*\* means  $p < 0.01$ , \*\*\* means  $p < 0.001$ .

Scenario	Approach	Backbone					
		EEGNet	DeepCNN	ShallowCNN	FBCNet	EEGConformer	Average acc±std
Within-subject	CSP-LR	-	-	-	-	-	74.95
	Standard	75.22±1.94	55.75±2.92	70.70±1.90	78.55±1.75	75.72±1.88	71.19±2.08
	CSP-Net-1-upd	<b>76.69</b> ±3.23	<b>61.03</b> ***±3.12	<b>73.12</b> ±2.02	<b>81.25</b> *±1.50	<b>81.07</b> ***±2.21	<b>74.6</b> ***±2.42
	CSP-Net-1-fix	<b>78.92</b> *±1.67	<b>61.33</b> ***±1.51	<b>74.89</b> *±1.84	<b>81.17</b> ±1.98	<b>80.96</b> ***±1.23	<b>75.45</b> ***±1.61
	CSP-Net-2-upd	<b>76.61</b> ±2.37	<b>56.84</b> ±1.19	<b>73.94</b> ***±1.35	<b>79.43</b> ±2.90	<b>79.96</b> *±2.32	<b>73.36</b> ***±2.03
	CSP-Net-2-fix	<b>80.01</b> ***±1.47	<b>66.81</b> ***±2.26	<b>77.14</b> ***±2.17	77.51±2.06	<b>78.64</b> ±3.25	<b>76.02</b> ***±2.25
Cross-subject	CSP-LR	-	-	-	-	-	74.21
	Standard	73.37±1.09	68.51±1.26	70.80±0.48	72.91±0.91	63.10±2.41	69.74±1.23
	CSP-Net-1-upd	73.19±1.34	<b>71.94</b> ***±1.26	<b>73.03</b> **±0.22	<b>73.08</b> ±0.93	<b>73.83</b> ***±0.61	<b>73.01</b> ***±0.76
	CSP-Net-1-fix	<b>76.60</b> ***±0.62	<b>71.56</b> ***±0.55	<b>73.87</b> ***±0.30	<b>73.91</b> ±0.55	<b>73.71</b> ***±0.47	<b>73.93</b> ***±0.49
	CSP-Net-2-upd	73.07±1.94	<b>70.04</b> ±1.28	69.67±1.43	<b>73.11</b> ±1.64	<b>70.93</b> ***±1.06	<b>71.36</b> ***±1.47
	CSP-Net-2-fix	<b>75.44</b> *±1.15	<b>71.06</b> ***±0.31	69.83±0.97	<b>72.64</b> ±0.70	<b>71.21</b> ***±0.48	<b>72.04</b> ***±0.72

Table 10: Average classification accuracies (%) on MI9S. Those higher than Standard are marked in bold. Asterisks indicate statistically significant differences between standard backbone and CSP-Net under adjusted paired *t*-test, where \* means  $p < 0.05$ , \*\* means  $p < 0.01$ , \*\*\* means  $p < 0.001$ .

Scenario	Approach	Backbone					
		EEGNet	DeepCNN	ShallowCNN	FBCNet	EEGConformer	Average acc±std
Within-subject	CSP-LR	-	-	-	-	-	67.84
	Standard	70.85±0.88	62.34±3.39	69.03±1.51	72.55±1.49	69.65±1.08	68.88±1.67
	CSP-Net-1-upd	<b>73.01</b> ±2.31	<b>62.92</b> ±1.53	<b>72.59</b> *±0.98	<b>73.70</b> ±1.12	<b>74.91</b> ***±2.22	<b>71.43</b> ***±1.63
	CSP-Net-1-fix	<b>73.63</b> *±1.85	<b>63.63</b> ±1.78	<b>73.17</b> ***±1.43	<b>74.31</b> ±2.17	<b>73.22</b> *±0.79	<b>71.59</b> ***±1.61
	CSP-Net-2-upd	<b>72.01</b> ±3.06	<b>62.78</b> ±3.09	<b>71.38</b> *±1.07	71.67±2.49	<b>73.55</b> *±2.07	<b>70.28</b> **±2.36
	CSP-Net-2-fix	<b>71.83</b> ±1.98	<b>65.57</b> ±0.96	67.19±1.68	72.41±2.57	<b>72.79</b> ±2.76	<b>69.96</b> ±1.99
Cross-subject	CSP-LR	-	-	-	-	-	57.72
	Standard	62.40±1.26	54.99±1.15	62.26±0.79	63.84±1.35	58.92±1.74	60.48±1.26
	CSP-Net-1-upd	<b>64.77</b> *±1.75	<b>59.49</b> **±2.51	61.73±0.70	63.41 ±0.69	<b>62.73</b> ***±0.92	<b>62.43</b> ***±1.31
	CSP-Net-1-fix	<b>64.91</b> *±2.28	<b>60.07</b> **±1.91	62.16±1.45	62.88 ±0.95	<b>63.11</b> ***±0.59	<b>62.63</b> ***±1.44
	CSP-Net-2-upd	62.03±1.76	<b>61.98</b> ***±0.95	<b>63.53</b> ±1.78	<b>63.54</b> ±1.15	<b>59.96</b> ±0.91	<b>62.21</b> **±1.31
	CSP-Net-2-fix	<b>64.72</b> ±0.59	<b>59.87</b> **±1.70	62.14±0.58	61.14±1.49	<b>59.96</b> ±0.69	<b>61.57</b> ±1.01

Table 11: Classification accuracies (%) for each model, averaged over all subjects.

Scenario	Dataset	CSP-LR	FBCSP-LR	MDRM	Deep-CNN	LMDA-Net	Shallow-CNN	EEGConformer	EEGNet	FBCNet	CSP-Net-1	CSP-Net-2
Within-subject	MI4C	61.43	66.81	64.21	51.51	56.75	65.13	61.15	63.50	65.30	71.98	71.82
	MI2C	75.72	76.72	76.32	61.46	71.19	76.29	75.31	76.38	78.11	81.06	81.53
	MI14S	74.95	75.39	78.58	55.75	72.04	70.70	75.72	75.22	78.55	81.07	79.96
	MI9S	67.84	69.37	67.90	62.34	69.94	69.03	69.65	70.85	72.55	74.91	73.55
Cross-subject	Average	69.99	72.07	71.75	57.77	67.48	70.29	70.46	71.49	73.63	76.26	75.85
	MI4C	52.55	49.72	51.08	47.38	48.28	52.58	40.21	49.44	46.52	52.17	51.72
	MI2C	72.92	73.62	71.99	73.15	69.10	74.32	65.25	71.50	69.34	74.51	75.25
	MI14S	74.21	76.36	71.71	68.51	67.87	70.80	63.10	73.37	72.91	76.60	75.44
	MI9S	57.27	63.61	57.39	54.99	63.41	62.26	58.92	62.40	63.84	64.91	64.72
	Average	64.24	65.83	63.04	61.01	62.17	64.99	56.87	64.18	63.15	67.05	66.78

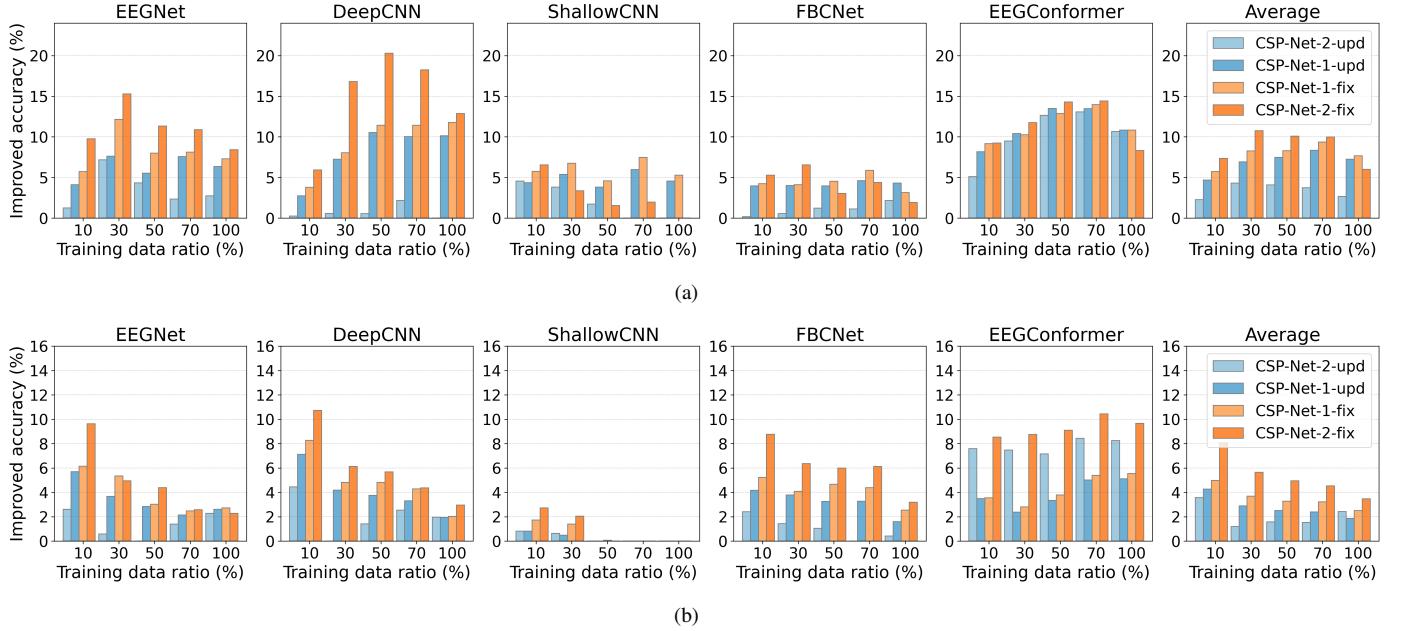


Figure 3: Accuracy improvements of CSP-Nets at different training data ratios on MI4C. (a) within-subject classification; and, (b) cross-subject classification.

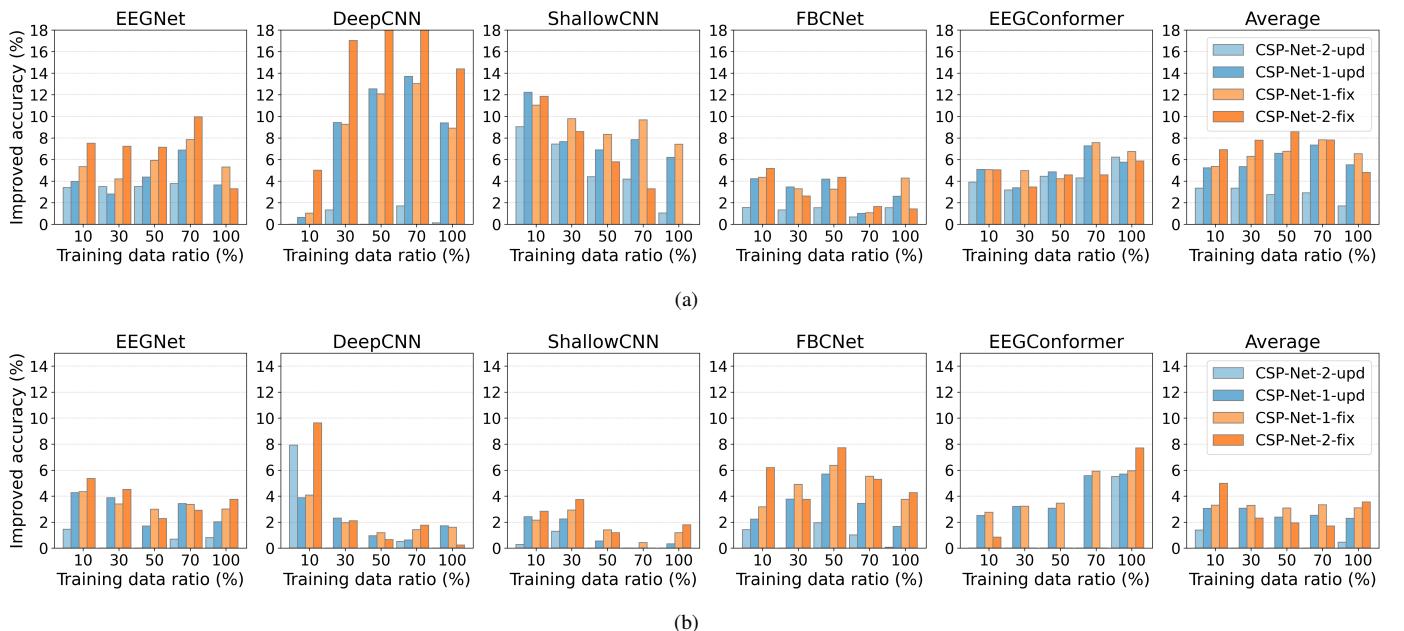
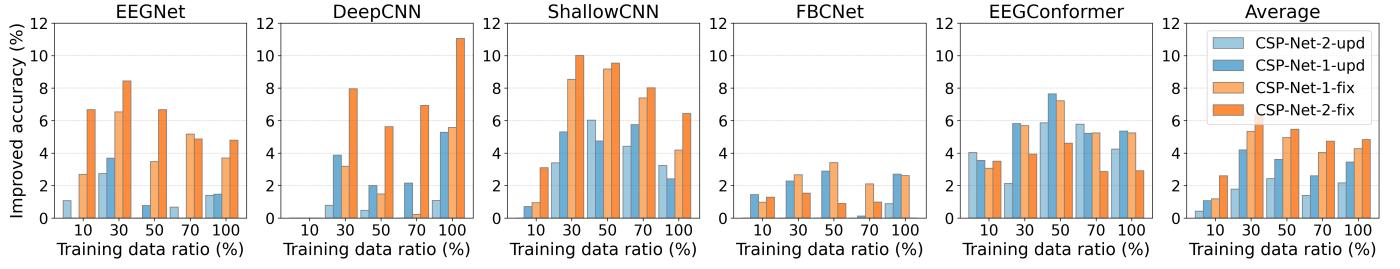
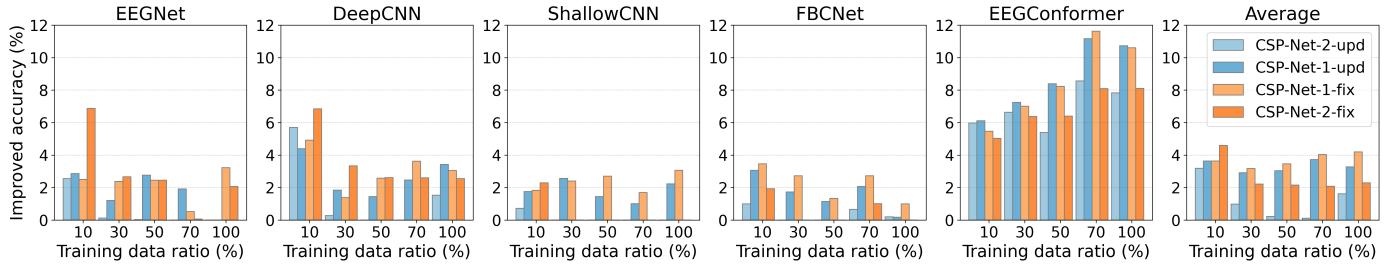


Figure 4: Accuracy improvements of CSP-Nets at different training data ratios on MI2C. (a) within-subject classification; and, (b) cross-subject classification.

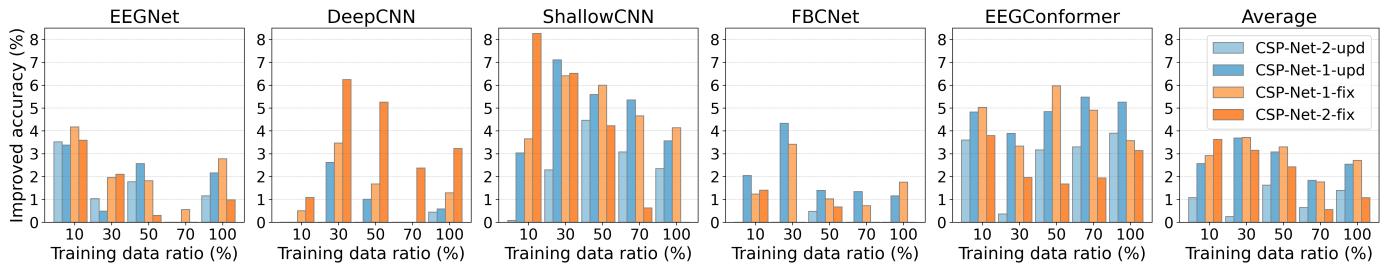


(a)

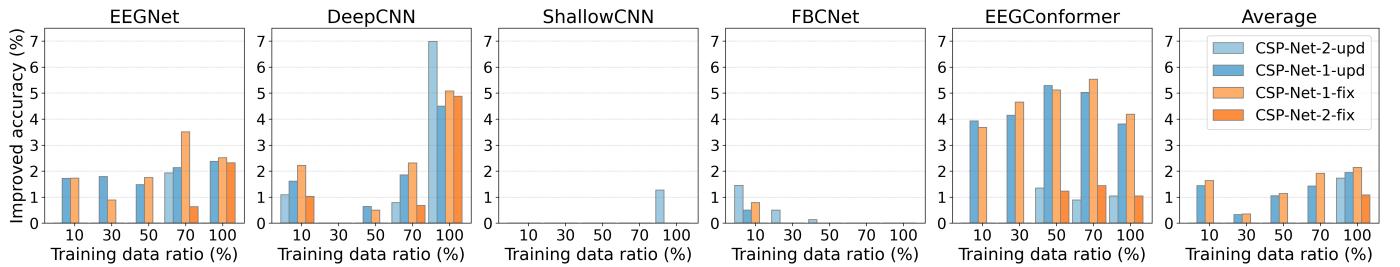


(b)

Figure 5: Accuracy improvements of CSP-Nets at different training data ratios on MI14S. (a) within-subject classification; and, (b) cross-subject classification.



(a)



(b)

Figure 6: Accuracy improvements of CSP-Nets at different training data ratios on MI9S. (a) within-subject classification; and, (b) cross-subject classification.

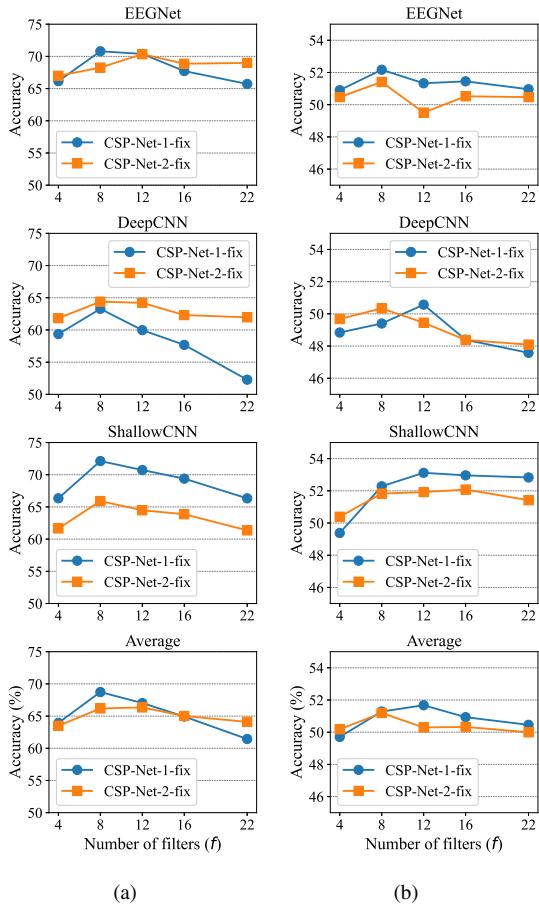


Figure 7: Classification accuracies of CSP-Nets using different number of CSP filters on MI4C. (a) within-subject classification; and, (b) cross-subject classification.

Table 12: Ablation study results of CSP-Net-1. Average accuracies higher than Standard are marked in bold.

Dataset	Backbone	Approach	Training data ratio (%)				
			10	30	50	70	100
MI4C	EEGNet	Standard	37.67	45.60	54.29	57.61	63.50
		CSP-Net-1-rad	37.60	47.19	54.37	57.62	65.60
		CSP-Net-1-fix	43.39	57.75	62.28	65.72	70.79
MI2C	DeepCNN	Standard	27.17	28.90	32.13	44.66	51.51
		CSP-Net-1-rad	27.33	31.40	32.84	47.53	53.51
		CSP-Net-1-fix	30.97	36.95	43.56	56.09	63.28
MI14S	ShallowCNN	Standard	34.98	48.69	55.42	59.01	65.13
		CSP-Net-1-rad	36.78	47.59	54.16	60.68	64.69
		CSP-Net-1-fix	40.73	55.44	60.00	66.47	70.42
MI9S	Average	Standard	33.27	41.06	47.28	53.76	60.05
		CSP-Net-1-rad	33.90	42.06	47.12	55.28	61.27
		CSP-Net-1-fix	<b>38.36</b>	<b>50.05</b>	<b>55.28</b>	<b>62.76</b>	<b>68.16</b>
MI2S	EEGNet	Standard	56.77	67.53	69.37	70.61	76.38
		CSP-Net-1-rad	57.77	66.76	68.51	71.85	77.11
		CSP-Net-1-fix	62.12	71.73	75.29	78.45	81.69
MI14S	DeepCNN	Standard	51.04	50.78	51.74	55.13	61.46
		CSP-Net-1-rad	51.97	52.50	54.14	56.94	62.69
		CSP-Net-1-fix	52.07	60.04	63.82	68.17	70.37
MI9S	ShallowCNN	Standard	51.38	62.34	68.36	71.70	76.29
		CSP-Net-1-rad	54.24	64.55	69.45	71.21	77.44
		CSP-Net-1-fix	62.42	72.12	76.69	81.37	83.71
MI14S	Average	Standard	53.06	60.22	63.16	65.81	71.38
		CSP-Net-1-rad	54.66	61.27	64.03	66.67	72.41
		CSP-Net-1-fix	<b>58.87</b>	<b>67.96</b>	<b>71.93</b>	<b>76.00</b>	<b>78.59</b>
MI2S	EEGNet	Standard	59.10	66.99	72.25	73.59	75.22
		CSP-Net-1-rad	57.55	68.00	71.30	71.79	74.40
		CSP-Net-1-fix	61.79	73.53	75.72	78.76	78.92
MI14S	DeepCNN	Standard	51.11	50.80	53.92	56.64	55.75
		CSP-Net-1-rad	49.64	53.58	55.95	57.04	55.99
		CSP-Net-1-fix	49.33	53.99	55.40	56.87	61.33
MI9S	ShallowCNN	Standard	53.05	57.86	62.07	65.42	70.70
		CSP-Net-1-rad	53.08	57.48	61.79	66.19	71.56
		CSP-Net-1-fix	54.00	66.40	71.24	72.82	74.89
MI2S	Average	Standard	54.42	58.55	62.75	65.22	67.22
		CSP-Net-1-rad	53.42	59.69	63.01	65.01	67.32
		CSP-Net-1-fix	<b>55.04</b>	<b>64.64</b>	<b>67.45</b>	<b>69.48</b>	<b>71.71</b>
MI9S	EEGNet	Standard	57.47	65.43	68.11	70.85	70.85
		CSP-Net-1-rad	59.03	64.89	67.19	69.97	69.57
		CSP-Net-1-fix	61.64	67.39	69.93	71.41	73.63
MI2S	DeepCNN	Standard	52.06	52.53	54.28	59.09	62.34
		CSP-Net-1-rad	51.93	53.01	54.22	56.45	61.92
		CSP-Net-1-fix	52.56	56.00	55.96	57.06	63.63
MI9S	ShallowCNN	Standard	51.89	57.55	61.90	66.04	69.03
		CSP-Net-1-rad	52.45	57.89	62.16	64.19	69.28
		CSP-Net-1-fix	55.54	63.96	67.90	70.70	73.17
MI2S	Average	Standard	53.81	58.50	61.43	65.33	67.41
		CSP-Net-1-rad	54.47	58.60	61.19	63.54	66.92
		CSP-Net-1-fix	<b>56.58</b>	<b>62.45</b>	<b>64.60</b>	<b>66.39</b>	<b>70.14</b>

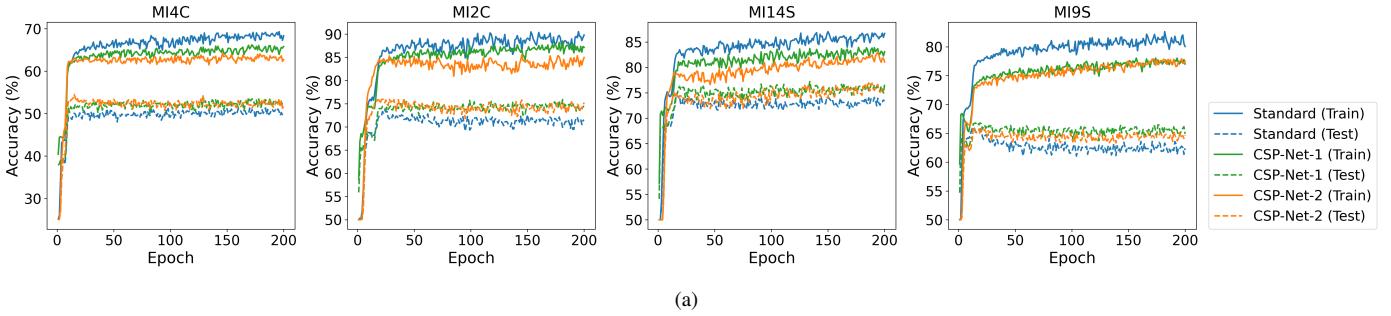


Figure 8: The training and test curves of different models on (a) MI4C; (b) MI2C; (c) MI14S; and, (d) MI9S.

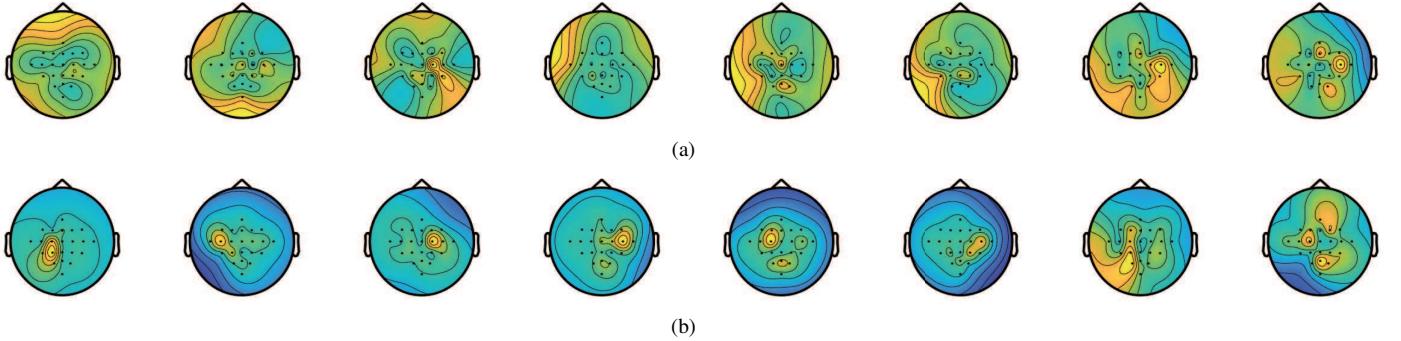


Figure 9: Visualization of eight (a) spatial filters in EEGNet and (b) the CSP filters in CSP-Net-2-fix for the within-subject classification on the Subject 1 of MI2C.

specific sensorimotor area, which aligned well with the spatial characteristics of MI. In contrast, the standard EEGNet struggled to learn effective spatial information due to the absence of prior knowledge provided by CSP. This alignment suggests that CSP filters effectively capture the relevant spatial patterns inherent in the EEG data, enhancing the interpretability of the model.

#### 4. Conclusions

Spatial information, which can be well captured by CSP filters, is critical in EEG-based MI classification. This paper has introduced two CSP-Nets, which integrate the knowledge-driven CSP filters with data-driven CNN models. CSP-Net-1 directly adds a CSP layer before a CNN, utilizing CSP-filtered signals as input to enhance the discriminability. CSP-Net-2 replaces a convolutional layer in CNN with a CSP layer. Experiments on four public MI datasets demonstrated that the two CSP-Nets consistently improved over their CNN backbones, in both within-subject and cross-subject classifications. They are particularly useful when the number of training samples is very small. Remarkably, CSP-Net-1-fix, whose CSP layer uses fixed weights calculated using the traditional CSP algorithm, is the simplest yet demonstrates overall best performance.

Our work demonstrates the advantage of integrating knowledge-driven CSP filters with data-driven CNNs, or traditional machine learning with deep learning, in EEG-based BCIs.

#### References

- [1] B. Graimann, B. Allison, G. Pfurtscheller, Brain-Computer Interfaces: A Gentle Introduction, Springer, Berlin, Heidelberg, 2009, pp. 1–27.
- [2] L. F. Nicolas-Alonso, J. Gomez-Gil, Brain computer interfaces, a review, Sensors 12 (2) (2012) 1211–1279.
- [3] L. R. Hochberg, D. Bacher, B. Jarosiewicz, N. Y. Masse, J. D. Simeral, J. Vogel, S. Haddadin, J. Liu, S. S. Cash, P. Van Der Smagt, et al., Reach and grasp by people with tetraplegia using a neurally controlled robotic arm, Nature 485 (7398) (2012) 372–375.
- [4] G. K. Anumanchipalli, J. Chartier, E. F. Chang, Speech synthesis from neural decoding of spoken sentences, Nature 568 (7753) (2019) 493–498.
- [5] M. Krauledat, K. Grzeska, M. Sagebaum, B. Blankertz, C. Vidaurre, K.-R. Müller, M. Schröder, Playing pinball with non-invasive BCI, Advances in Neural Information Processing Systems 21 (2008) 1641–1648.
- [6] G. Pfurtscheller, C. Neuper, Motor imagery and direct brain-computer communication, Proc. of the IEEE 89 (7) (2001) 1123–1134.
- [7] M. Jeannerod, Mental imagery in the motor context, Neuropsychologia 33 (11) (1995) 1419–1432.
- [8] G. Pfurtscheller, C. Neuper, D. Flotzinger, M. Pregenzer, EEG-based discrimination between imagination of right and left hand movement, Electroencephalography and Clinical Neurophysiology 103 (6) (1997) 642–651.
- [9] B. Blankertz, C. Sannelli, S. Halder, E. M. Hammer, A. Kübler, K.-R. Müller, G. Curio, T. Dickhaus, Neurophysiological predictor of SMR-based BCI performance, NeuroImage 51 (4) (2010) 1303–1309.
- [10] H. Ramoser, J. Müller-Gerking, G. Pfurtscheller, Optimal spatial filtering of single trial EEG during imagined hand movement, IEEE Trans. on Neural Systems and Rehabilitation Engineering 8 (4) (2000) 441–446.
- [11] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, K.-r. Müller, Optimizing spatial filters for robust EEG single-trial analysis, IEEE Signal Processing Magazine 25 (1) (2008) 41–56.
- [12] G. Dornhege, B. Blankertz, G. Curio, K.-R. Müller, Boosting bit rates in noninvasive EEG single-trial classifications by feature combination and multiclass paradigms, IEEE Trans. on Biomedical Engineering 51 (6) (2004) 993–1002.
- [13] K. K. Ang, Z. Y. Chin, H. Zhang, C. Guan, Filter bank common spatial

- pattern (FBCSP) in brain-computer interface, in: Proc. IEEE Int'l Joint Conf. on Neural Networks, Hong Kong, China, 2008, pp. 2390–2397.
- [14] F. Lotte, C. Guan, Regularizing common spatial patterns to improve BCI designs: unified theory and new algorithms, *IEEE Trans. on Biomedical Engineering* 58 (2) (2010) 355–362.
- [15] A. Craik, Y. He, J. L. Contreras-Vidal, Deep learning for electroencephalogram (EEG) classification tasks: a review, *Journal of Neural Engineering* 16 (3) (2019) 031001.
- [16] H. Altaheri, G. Muhammad, M. Alsulaiman, S. U. Amin, G. A. Altwaijri, 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.
- [17] A. Al-Saegh, S. A. Dawwd, J. M. Abdul-Jabbar, Deep learning for motor imagery EEG-based classification: A review, *Biomedical Signal Processing and Control* 63 (2021) 102172.
- [18] 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 (11) (2017) 5391–5420.
- [19] 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.
- [20] 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).
- [21] Y. Song, Q. Zheng, B. Liu, X. Gao, EEG Conformer: Convolutional transformer for EEG decoding and visualization, *IEEE Trans. on Neural Systems and Rehabilitation Engineering* 31 (2023) 710–719.
- [22] Z. J. Koles, M. S. Lazar, S. Z. Zhou, Spatial patterns underlying population differences in the background EEG, *Brain Topography* 2 (1990) 275–284.
- [23] J. Müller-Gerking, G. Pfurtscheller, H. Flyvbjerg, Designing optimal spatial filters for single-trial EEG classification in a movement task, *Clinical Neurophysiology* 110 (5) (1999) 787–798.
- [24] M. Tangermann, K.-R. Müller, A. Aertsen, N. Birbaumer, C. Braun, C. Brunner, R. Leeb, C. Mehring, K. Miller, G. Mueller-Putz, G. Nolte, G. Pfurtscheller, H. Preissl, G. Schalk, A. Schlogl, C. Vidaurre, S. Waldert, B. Blankertz, Review of the BCI Competition IV, *Frontiers in Neuroscience* 6 (2012) 55.
- [25] J. Faller, C. Vidaurre, T. Solis-Escalante, C. Neuper, R. Scherer, Autocalibration and recurrent adaptation: Towards a plug and play online ERD-BCI, *IEEE Trans. on Neural Systems and Rehabilitation Engineering* 20 (3) (2012) 313–319.
- [26] K. Xia, L. Deng, W. Duch, D. Wu, Privacy-preserving domain adaptation for motor imagery-based brain-computer interfaces, *IEEE Trans. on Biomedical Engineering* 69 (11) (2022) 3365–3376.
- [27] V. Jayaram, A. Barachant, MOABB: trustworthy algorithm benchmarking for BCIs, *Journal of Neural Engineering* 15 (6) (2018) 066011.
- [28] A. Barachant, S. Bonnet, M. Congedo, C. Jutten, Multiclass brain-computer interface classification by Riemannian geometry, *IEEE Trans. on Biomedical Engineering* 59 (4) (2012) 920–928.
- [29] 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* (2023) 120209.