IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 13, 2021, accepted January 20, 2021, date of publication February 1, 2021, date of current version February 12, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3056088

Advanced TSGL-EEGNet for Motor Imagery EEG-Based Brain-Computer Interfaces

XIN DENG[1], (Member, IEEE), BOXIAN ZHANG[1], NIAN YU[2], KE LIU[1], AND KAIWEI SUN[1]

[1]Key Laboratory of Data Engineering and Visual Computing, College of Computer Science and Technology, Chongqing University of Posts and Telecommunication, Chongqing 400065, China
[2]School of Electrical Engineering, Chongqing University, Chongqing 400044, China

Corresponding author: Nian Yu (yunian@126.com)

This work was supported in part by the Natural Science Foundation of Chongqing under Grant cstc2020jcyj-msxmX0284, in part by the Scientific and Technological Research Program of Chongqing (CQ) Municipal Education Commission under Grant KJQN202000625, in part by the National Nature Science Foundation of China under Grant 61806033 and Grant 61703065, and in part by the Key Industry Core Technology Innovation Project of CQ under Grant cstc2017zdcy-zdyfX0012.

ABSTRACT Deep learning technology is rapidly spreading in recent years and has been extensive attempts in the field of Brain-Computer Interface (BCI). Though the accuracy of Motor Imagery (MI) BCI systems based on the deep learning have been greatly improved compared with some traditional algorithms, it is still a big problem to clearly interpret the deep learning models. To address the issues, this work first introduces a popular deep learning model EEGNet and compares it with the traditional algorithm Filter-Bank Common Spatial Pattern (FBCSP). After that, this work considers that the 1-D convolution of EEGNet can be explained by a special Discrete Wavelet Transform (DWT), and the depthwise convolution of EEGNet is similar to the Common Spatial Pattern (CSP) algorithm. Therefore, this work improves the EEGNet by using the algorithm Temporary Constrained Sparse Group Lasso (TCSGL) to enhance its performance. The proposed model TSGL-EEGNet is tested on the BCI Competition IV 2a and BCI Competition III IIIa datasets that both are 4-classes classification MI tasks. The testing results show that the proposed model has achieved 78.96% (0.7194) average classification accuracy (kappa) on the dataset BCI Competition IV 2a, which are greater than EEGNet, C2CM, MB3DCNN, SS-MEMDBF and FBCSP, especially on insensitive subjects. The proposed model has also achieved 85.30% (0.8040) average classification accuracy (kappa) on the dataset BCI Competition III IIIa, which are greater than the EEGNet, MFTFS *et al.* At last, this work uses average-validation and stacking to further enhance the effect of the model. The 4-classes classification average accuracy rates reach 81.34% and 88.89%, and the kappas reach 0.7511 and 0.8519 on dataset BCI Competition IV 2a and BCI Competition III IIIa, respectively. Additionally, this work also uses the Grad-CAM to visualize the frequency and spatial features that are learned by the neural network.

INDEX TERMS Motor imagery, BCI, CNN, FBCSP, temporary constrained sparse group Lasso.

I. INTRODUCTION

To decode the MI EEG precisely is one of the key issues for the BCI system. Generally, BCI system includes three aspects. The first one is the signal processing and data enhancement. The second one is feature extraction, including feature selection and fusion, and the last one is the classification and recognition. Generally speaking, the former process is called signal pre-processing, the latter two processes are called signal decoding [1]–[8]. Alternatively, the second process is also called the decoding, and the last process is called the classification. In the traditional machine learning methods, motor imagery based BCI system mainly monitors sensorimotor rhythm (SMR). The SMR is an oscillatory rhythm in electrical brain signals that originates in brain regions involved in the preparation, control and execution of voluntary movements [9], [10]. The increase in activity in a particular frequency band is called event-related synchronization (ERS), while the decrease in a particular frequency band is called event-related desynchronization (ERD) [9]. The motor imagery, motor activity and sensory stimulation can trigger the ERSs and ERDs [11], [12]. The EEG for

The associate editor coordinating the review of this manuscript and approving it for publication was Vincent Chen.

the brain activity is usually divided into five distinct types: δ rhythm ($<$ 4 Hz), θ rhythm (4-7 Hz), α rhythm (8-12 Hz), β rhythm (12-30 Hz), γ rhythm ($>$ 30 Hz), and so on. In references [4], [13], the α rhythm recorded from within the sensorimotor area of the cerebral cortex is called μ rhythm. For the left-hand and right-hand classification of motor imagery tasks, μ and β rhythm variations are used to identify the type of task in progress. The γ rhythm is reliably used in invasive MI BCI, but it is rarely used effectively in scalp EEG. For the multi-object MI BCI system, the common motion types include the left hand, right hand, foot and tongue motion [14]–[16]. These events have been shown to produce significant discriminative changes in the EEG signal relative to the background EEG signal. Additionally, the feet motions are usually grouped together, and there is no distinction between left and right foot motion, as well as the movements of specific fingers. This is because the cortical areas associated with these different movements are too small to produce distinctive ERD and ERS signals [9]. However, as far as we know, there is only one research [17] suggests that β rhythm has the potential to be used to distinguish imagery signals of movement between the left and right foot.

To process the EEG signals is challengeable because they are non-stationary, and easily affected by external noise and prone to signal camouflage. Additionally, the EEG acquisition is difficult, and the signal-to-noise ratio (SNR) is low, and the size of EEG dataset (number of trials) is usually small. In MI tasks, although the number of target samples of each classification is relatively balanced, there are difficulties in persons sensitivity. For example, based on the estimation of the classification effect of the state-of-the-art model on BCI Competition IV 2a dataset, about 30% of participants were not sensitive to MI tasks (i.e. the effect of the same decoding method varies from person to person). Furthermore, the EEG signals were also influenced by the subjects' postures and emotions. In recent years, there has been a lot of research on MI BCI. Ang *et al.* [18] proposed a classical algorithm FBCSP. It should be noted that the FBCSP has the mathematical derivation and good interpretability. It won the championship of BCI Competition IV 2a/b dataset in 2008, but it is not enough effective in practice. Schirrmeister *et al.* [6] proposed the Deep Conv and Shallow Conv based on convolutional neural network (CNN). The Deep Conv and Shallow Conv use one-dimensional (1-D) convolution to extract effective information, and explore the convolution structure and the effectiveness of deep learning in MI BCI system. Lawhern *et al.* [3] proposed the EEGNet, a universal deep learning framework for the EEG tasks. The EEGNet is a common Deep learning framework for multiple EEG paradigm, which is based on the improvement of the Deep Conv and Shallow Conv, and achieves higher decoding accuracy and shorter training time. Zhao *et al.* [7] used a multi-branch 3D CNN method to classify MI EEG signals. Their method was somewhat like the popular feature pyramid networks for object detection in videos. However, because EEG signals are time-varying and non-stationary, the 3D CNN cannot perform as well as in the object detection task. Ha and Jeong [8] used CapsuleNet to realize a MI BCI system. They transformed EEG into graphs, and this transformation by using the Short-time Fourier Transform (STFT) might lose some EEG information. To avoid this problem, they use the CapsuleNet to process graph classification. Though the CapsuleNet was a good model in computer vision, some modifications and improvements should be applied to it if we use it in the BCI system. Generally speaking, some traditional algorithms have good interpretability, and their computational speeds are also fast. The neural network algorithms often get greater accuracy rate, but the training speeds are slow. It is not easy to be understood how these neural networks to achieve the results. Therefore, the main research of this paper is to make the neural network algorithm close to the traditional algorithm in interpretability, and keep or even improve its accuracy.

This paper proposes the Temporal-constrained Group Lasso EEGNet (TSGL-EEGNet) algorithm by using the regularization method, which is an convolutional neural network based approach for the motor imagery BCI system. Additionally, this method is based on the FBCSP and EEGNet, and improves them according to the Temporary Constrained Sparse Group Lasso (TCSGL) in [19]–[21]. The main work of the method proposed in this paper is to combine the traditional method and the deep learning models with the theory of MI BCI, and to discuss the interpretability of the neural network methods in BCI domain.

The rest of the paper is below. In Section II, we introduce the dataset and pre-processing method. In Section III, we review the related works, and present the mathematical formulation of our TSGL-EENet. In Section IV, we thoroughly evaluate the proposed model performance and use the visualization way to interpret it. After that, we make some discussions about the persons sensitivity and the method of using cropped training to select the optimal time-segment in Section V. Finally, we conclude the paper and propose a methodology to guide the building of neural network structures in future work.

II. DATA

In our work we use two public datasets. The first one is the BCI Competition IV 2a dataset (2008) [22], which is a classical dataset and involves 4 classes of motor imagery samples of left hand, right hand, feet and tongue movements from 9 subjects. The data were recorded by 22 Ag/AgCl electrodes, sampled at 250 Hz and bandpass filtered between 0.5 and 100 Hz and processed by 50 Hz notch filtering. One trial of the data were 4 seconds length, and it was obtained after 2 seconds fixation cross. The second dataset is the BCI Competition III IIIa, which contends 4 classes of motor imagery EEG signals of left hand, right hand, feet and tongue movements from 3 subjects. The data were recorded by 60 electrodes, sampled at 250 Hz and bandpass filtered between 1 and 50 Hz with Notchfilter on. One trial of the data were 4 seconds length, and it was obtained after 2 seconds blank screen and 1 seconds fixation cross with a short beep. In data pre-procession, the data with missing values are processed by using linear interpolation, detrending, filtering between 4 and 38 Hz and standardization. The EEG data in the datasets do not require some special data pre-processing measures, and some pre-processing measurements are from existing studies [3], [6], [7], such as detrending, standardization, filter and the method of removing artifact signals.

III. METHODS

A. FBCSP

The FBCSP algorithm is an excellent traditional algorithm, and its predecessor CSP algorithm is a kind of data-driven algorithm by learning a spatial filter (linear transformation) to maximize the two kinds of variance of training data for classification. Spatial filter is mainly concerned with the channel of EEG data. In the Motor Imagery paradigm, the left-hand and right-hand motor imagery responses have different channel responses, so the CSP algorithm achieves better performance in two-categories-classification tasks. The FBCSP adds the feature of frequency domain to CSP algorithm, which has been tested for a long time. The core of the FBCSP algorithm is described as follows [5], [18], [23]:

1) All EEG data are filtered by filter banks, which are usually composed of 4-8 Hz, 8-12 Hz, $\cdots$, 32-36 Hz. Usually, the bandwidth is 4 Hz, and the number of filters is 9.
2) After filtering, all the data is decomposed into nine bands. The data on each band are calculated using the CSP algorithm. This are done by maximizing the following objective function as Eq. (1):

$$\boldsymbol{w}^* = \arg\max_{\boldsymbol{w}} \frac{\boldsymbol{w}^T \sum_{c_1} \boldsymbol{w}}{\boldsymbol{w}^T \sum_{c_1} + \sum_{c_2} \boldsymbol{w}}, \tag{1}$$

where $\sum_{c_1}$ and $\sum_{c_2}$ correspond to the channel covariance matrix of $\sum_{c_1}$ and $\sum_{c_2}$, respectively. $\boldsymbol{w}$ is a spatial filter. This objective function, also known as a Rayleigh Quotient, has an analytic solution, which is equivalent to solving a generalized eigenvalue decomposition (GEVD) problem. If $\boldsymbol{x}$ is the sample, i is the class number, and j is the frequency band ordinal, then the feature $\boldsymbol{\mathcal{F}}$ is derived from the following function Eq. (2):

$$\boldsymbol{\mathcal{F}}_{i,j} = \boldsymbol{w}\boldsymbol{x}_{i,j}. \tag{2}$$

3) The $2 \times NW$ extreme eigenvalues (the maximum and the minimum of NW eigenvalues) corresponding to the spatial filter are selected. Then each maximum and minimum eigenvalues of the spatial filter is matched to each other accordingly (spatial filter channel pair).
4) The energy (variance) of the spatially filtered channel is calculated and normalized to the total energy of the channel in a given frequency band. The logarithm of the energy is the final features.
5) By concatenating the features from all 9 filtering bands, the mutual information-based feature selection is carried out on $2 \times NW \times 9$ spatial filtering channels, and NS filtering channel pairs are selected. A maximum of $2 \times NS$ features can be selected based on whether the selected features are already paired with each other.
6) Because CSP is designed for two-classes classification problems, in the case of multi-classes tasks, a one-vs-rest or one-vs-one policy must be specified. The former will result in up to $class \times 2 \times NS$ features in FBCSP.
7) SVM and other maximum interval classification algorithms are used to effectively classify the features.

B. EEGNet

The EEGNet imitates the feature engineering of FBCSP in some way. It consists of a 1-D convolution, a depth-wise convolution and a separable convolution. The final classification is carried out through full connection layer and activated by the Softmax function.

In Fig. 1, the structure of EEGNet is represented as three parts: feature extraction, feature selection and classification. All of its convolution structures do not use the bias terms. There are several reasons to say that it imitates the FBCSP's feature engineering. In feature extraction, the EEGNet consists of a 1-D convolution and a depth-wise convolution.

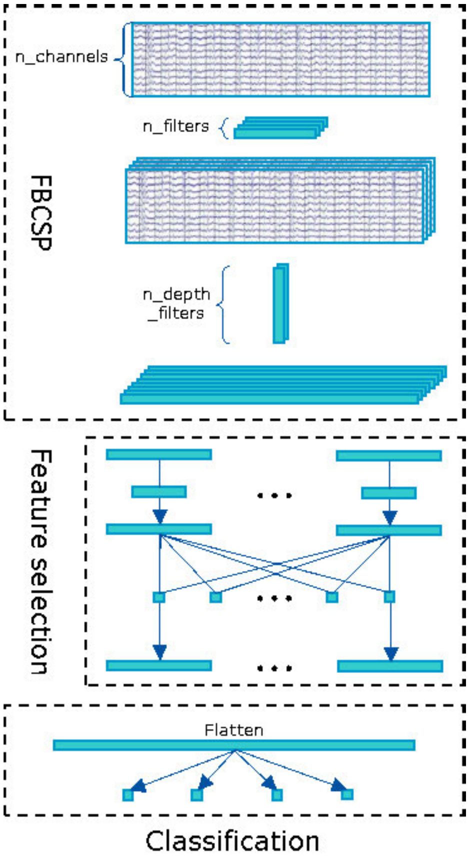

FIGURE 1. The structure of EEGNet.

The essence of convolution is to calculate the similarity between the convolution and the convolution kernel. This is similar to the DWT process, and the convolution kernel is similar to the wavelet basis function with a fixed scale. Actually, the DWT can be realized by the filter bank. Hence, the 1-D convolution without the bias is a filter bank for a signal, and the convolution kernel is the filter. Therefore, when 1-D convolution kernel convolutes along the time dimension for a single channel, it can be regarded as filtering the frequency. The depth-wise convolution convolutes the data on the channel dimension, and obtains a set of spatial filters. Each spatial filter represents a linear transformation that maps all channels to one feature. The whole procedure is similar to the FBCSP algorithm.

The EEGNet's feature extraction can be expressed by the following formula Eq. 3:

$$\mathcal{F}_{i,j} = \boldsymbol{w}(\boldsymbol{f}_j \boldsymbol{x}_i) = \boldsymbol{w}\boldsymbol{x}_{i,j}, \tag{3}$$

where $\mathcal{F}$ is the features, and $\boldsymbol{x}$ is the samples, and $\boldsymbol{w}$ is the spatial filters, and $\boldsymbol{f}$ is the frequency filters. i and j represent the category number and the frequency filter ordinal, respectively. It can be seen that Eq. (2) is equivalent to Eq. (3) in general, and the latter has more variable parameters and higher degrees of freedom. Together, the feature learning part of the EEGNet consists of a filter bank (1-D convolution) and a spatial filter (depth-wise convolution), which is the main content of the FBCSP.

In feature selection, it convolutes time-spatial features in different frequencies with 1 × 16 depth-wise convolution firstly and then using 1 × 1 point-wise convolution to mix them, which cannot be exactly understood. As a fact, the separable convolution is not exactly choosing features but mixing them with different weights to imitate the selection. In practical application, the feature selection part increases the robustness of the EEGNet and improves the accuracy of decoding and classification. Feature selection is also a very important part of CSP algorithms. The feature selection methods commonly used in MI BCI [18], [23] such as the Mutual Information based Best Individual Feature (MIBIF), the Mutual Information-based Naïve Bayesian Parzen Window (MINBPW), the Mutual Information based feature selection (MIFS), the Fuzzy-Rough set-based Feature Selection (FRFS), the Mutual Information-based Rough Set Reduction (MIRSR). The regularization algorithms are also used as feature selection methods recently by Zhang *et al.* [19], and Jin *et al.* [24], [25]. Choosing the appropriate feature selection method is a good way to improve the model effect.

At last, in the classification part, its input includes three domains: time, frequency and spatial domains. The time domain features are generated by time dimension. The frequency domain features are generated by 1-D convolution and separable convolution. The spatial domain features are generated by depth-wise convolution based on frequency domain features. After all these features are selected, four probabilities are output through a fully connected layer with the activation function of Softmax to get the category.

By conclusion, the main ideas of the EEGNet and FBCSP algorithm are the same, which can be said that the EEGNet is a deep learning algorithm evolved from the classic algorithm.

C. TSGL-EEGNet

In the field of deep learning technology of BCI, EEGNet is a delicate algorithm, which has a clear explanation. But it is still not enough, since its feature selection part is difficult to be understood, and the increasing of the feature space will lead to over-fitting. Keeping the main body of the EEGNet unchanged, increasing the interpretability of the feature selection part and reducing the over-fitting cases will be a good beginning to solve the problem. To deal with the over-fitting, the regularization is a good method to solve the problem, and it also has good interpretability. In addition, the regularization is also one of the feature selection methods. Therefore, this paper proposes a Temporal-constrained Sparse Group Lasso EEGNet (TSGL-EEGNet) using regularization.

The structure of the TSGL-EEGNet is shown in Fig. 2 and Table 1. The main difference from EEGNet is the Feature selection section. Unlike the EEGNet's separable convolution, only 1-D convolution is used here for better interpretability. The TSGL penalty is added to this convolution, which makes the weight matrix of this layer change in the direction of penalty minimization during the training.

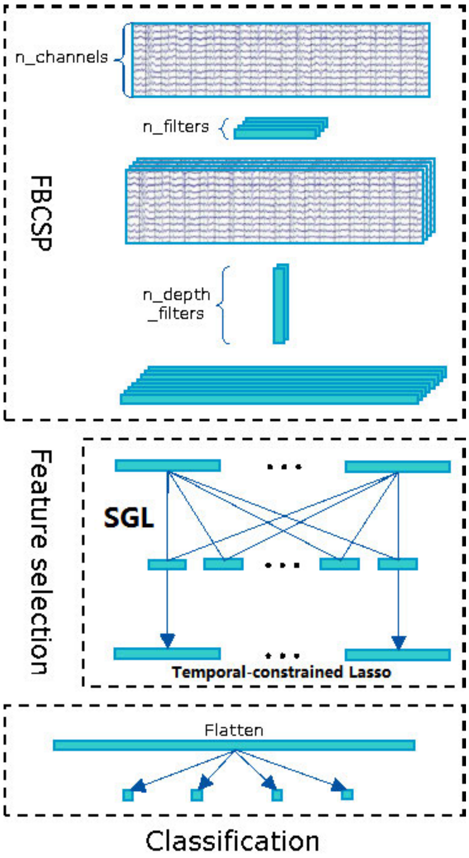

FIGURE 2. The structure of TSGL-EEGNet.

TABLE 1. Structure of TSGL-EEGNet.

Layer type	Output shape	Kernel size	Kernel numbers	Stride	Padding	Regularization	Activation
Input	$(N_{batch}, N_{channel}, N_{timestep}, 1)$	-	-	-	-	-	-
Convolution 2D	$(N_{batch}, N_{channel}, N_{timestep}, 16)$	$(1, 64)$	16	1	same	-	-
BatchNormalization	$(N_{batch}, N_{channel}, N_{timestep}, 16)$	-	-	-	-	-	-
DepthwiseConv 2D	$(N_{batch}, 1, N_{timestep}, 16 \times 10)$	$(N_{channel}, 1)$	10	1	-	-	-
BatchNormalization	$(N_{batch}, 1, N_{timestep}, 16 \times 10)$	-	-	-	-	-	-
Activation	$(N_{batch}, 1, N_{timestep}, 160)$	-	-	-	-	-	elu
AveragePooling 2D	$(N_{batch}, 1, \frac{N_{timestep}}{4}, 160)$	$(1, 4)$	-	-	-	-	-
Dropout	$(N_{batch}, 1, \frac{N_{timestep}}{4}, 160)$	-	-	-	-	-	-
Convolution 2D	$(N_{batch}, 1, \frac{N_{timestep}}{4}, 160)$	$(1, 16)$	160	1	same	TSGL	-
BatchNormalization	$(N_{batch}, 1, \frac{N_{timestep}}{4}, 160)$	-	-	-	-	-	-
Activation	$(N_{batch}, 1, \frac{N_{timestep}}{4}, 160)$	-	-	-	-	-	elu
AveragePooling 2D	$(N_{batch}, 1, \frac{N_{timestep}}{32}, 160)$	$(1, 8)$	-	-	-	-	-
Dropout	$(N_{batch}, 1, \frac{N_{timestep}}{32}, 160)$	-	-	-	-	-	-
Flatten	$(N_{batch}, \frac{N_{timestep}}{32} \times 160)$	-	-	-	-	-	-
Dense (FC)	$(N_{batch}, N_{classes})$	-	$N_{classes}$	-	-	-	-
Activation (Output)	$(N_{batch}, N_{classes})$	-	-	-	-	-	softmax

The TSGL regularization is shown in Fig. 3. The black squares represent the inhibition of parameters, and the white squares represent the activation, and the gray squares also represent the activation but with small weights. The Group Lasso (GL) can inhibit groups which is represented by outgoing vectors. The Sparse Group Lasso (SGL) can inhibit some parameters in the activated groups. The Temporal-constrained Lasso (TL) can keep temporal domain being smooth. The GL will inhibit or activate a whole group when it is affected, and the SGL will inhibit some parameters of the activated groups based on the GL, and the TL will decrease or increase some parameters to keep the difference of a group in temporal domain being small. In Fig. 3, it can be seen that the TSGL regularization consists of the TL and SGL. In order to explain these Lassos in mathematics, some symbols are defined as follow. In the N classification problem, for one sample, the selected feature can be expressed as $\boldsymbol{w}\mathcal{F}$, where the feature obtained by the feature learning part is a 2-D matrix named $\mathcal{F}$ and the 1-D convolution weight matrix is a 3-D matrix named $\boldsymbol{w}$.

The purpose of the Temporary Constraint is to keep the time domain features smooth and reduce the distortion caused by other regularization, so as to be closer to the real EEG signals. The Temporary Constraint uses the features of the latter time minus the features of the former time. Thus, when the temporal domain is smooth enough, the temporary loss should be close to 0. By assuming that there are T timesteps in EEG data, the features of 1 to $T-1$ timesteps are $\boldsymbol{w}\mathcal{F}_1$ and 2 to T timesteps are $\boldsymbol{w}\mathcal{F}_2$, so the Temporal-constrained Lasso can be expressed as $\|\boldsymbol{w}\mathcal{F}_2 - \boldsymbol{w}\mathcal{F}_1\|_1$.

The Sparse Group Lasso, which consists of a Group Lasso and a L1 Norm, can make the groups sparse. In the Sparse Group Lasso, the activated group elements can also have the sparsity. The frequency and spatial domain features are grouped in this paper to select appropriate frequency and spatial features, and the time domain features of these can also be selected by L1 Norm, which can be expressed as $\|\boldsymbol{w}\|_{2,1} + \|\boldsymbol{w}\|_1$. $\|\boldsymbol{w}\|_{2,1} = \sum_{\boldsymbol{g}\in\boldsymbol{w}} \sqrt{|\boldsymbol{g}|}\|\boldsymbol{g}\|_2 = \sum_{\boldsymbol{g}\in\boldsymbol{w}} \sqrt{|\boldsymbol{g}|\sum \boldsymbol{g}^2}$, $\|\boldsymbol{w}\|_1 = \sum_{\boldsymbol{w}} |\boldsymbol{w}|$, where $\boldsymbol{g}$ is a group vector that is generally

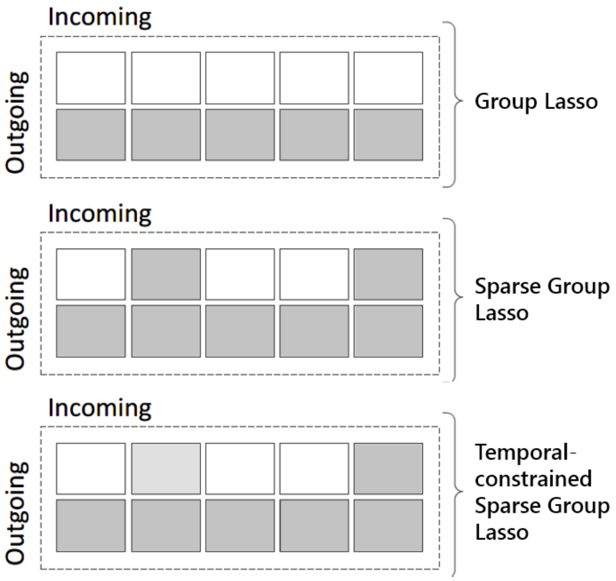

FIGURE 3. The GL, SGL and TSGL regularization. The black squares represent the inhibition of parameters, and the white squares represent the activation, and the gray squares also represent the activation but with small weights. Group Lasso (GL) can inhibit groups which is represented by outgoing vectors. Sparse Group Lasso (SGL) can inhibit some parameters in the activated groups. Temporal-constrained Lasso (TL) can keep temporal domain being smooth.

the length of a dimension of $\boldsymbol{w}$. $|\boldsymbol{g}|$ is the length of the vector. In this paper, we treat the features calculated by the feature selection convolution as a group, and each feature is an element of the group vector. Therefore $|\boldsymbol{g}|$ represents the length of the dimension that determines the number of output features in the matrix $\boldsymbol{w}$.

The final weight matrix $\boldsymbol{w}^*$ is obtained by solving the following problem Eq. (4):

$$\begin{aligned}\boldsymbol{w}^* = \; & \arg\min_{\boldsymbol{w},\boldsymbol{a},\boldsymbol{b}} -\frac{1}{n}\sum_{\boldsymbol{x}}\sum_{i=1}^{N}\big[\mathrm{Softmax}(\boldsymbol{aw}\mathcal{F}+\boldsymbol{b})_i\boldsymbol{y}_i \\ & +(1-\boldsymbol{y}_i)\ln(1-\mathrm{Softmax}(\boldsymbol{aw}\mathcal{F}+\boldsymbol{b})_i)\big] \\ & +\frac{\beta_1}{n}\|\boldsymbol{w}\|_{2,1}+\frac{\beta_2}{n}\|\boldsymbol{w}\|_1+\frac{\beta_3}{n}\|\boldsymbol{w}\mathcal{F}_2-\boldsymbol{w}\mathcal{F}_1\|_1, \quad (4)\end{aligned}$$

where $\beta_1, \beta_2, \beta_3$ are the regularization coefficients, and $\boldsymbol{x}$ is the sample, and $\boldsymbol{y}$ is the ground truth. $\boldsymbol{a}, \boldsymbol{b}$ is the Layer FC's weight and bias, respectively. n is the number of samples. Let σ represent the activation function (here is Softmax) and $\boldsymbol{z}$ represent $\boldsymbol{aw}\mathcal{F}+\boldsymbol{b}$, the problem's loss function $\mathscr{C}$ can be expressed as Eq. (5):

$$\begin{aligned}\mathscr{C} = & -\frac{1}{n}\sum_{\boldsymbol{x}}\sum_{i=1}^{N}\big[\sigma(\boldsymbol{z})_i\boldsymbol{y}_i+(1-\boldsymbol{y}_i)\ln(1-\sigma(\boldsymbol{z})_i)\big] \\ & +\frac{\beta_1}{n}\|\boldsymbol{w}\|_{2,1}+\frac{\beta_2}{n}\|\boldsymbol{w}\|_1+\frac{\beta_3}{n}\|\boldsymbol{w}\mathcal{F}_2-\boldsymbol{w}\mathcal{F}_1\|_1. \quad (5)\end{aligned}$$

The gradient descent method is used to obtain the approximate solution of the problem, where the gradient is Eq. (6), Eq. (7), and Eq. (8):

$$\begin{aligned}\frac{\partial\mathscr{C}}{\partial\boldsymbol{w}} &= \frac{1}{n}\Big[\sum_{\boldsymbol{x}}\sum_{i=1}^{N}\boldsymbol{a}\mathcal{F}\big(\sigma(\boldsymbol{z})_i-\boldsymbol{y}_i\big)+\beta_1\sqrt{|\boldsymbol{g}|}\frac{\boldsymbol{w}}{\|\boldsymbol{w}\|_2} \\ &\quad +\beta_2\mathrm{sgn}(\boldsymbol{w})+\beta_3\mathrm{sgn}(\boldsymbol{w}\mathcal{F}_2-\boldsymbol{w}\mathcal{F}_1)\Big], \quad (6) \\ \frac{\partial\mathscr{C}}{\partial\boldsymbol{a}} &= \frac{1}{n}\sum_{\boldsymbol{x}}\sum_{i=1}^{N}\boldsymbol{w}\mathcal{F}\big(\sigma(\boldsymbol{z})_i-\boldsymbol{y}_i\big), \quad (7) \\ \frac{\partial\mathscr{C}}{\partial\boldsymbol{b}} &= \frac{1}{n}\sum_{\boldsymbol{x}}\sum_{i=1}^{N}\big(\sigma(\boldsymbol{z})_i-\boldsymbol{y}_i\big). \quad (8)\end{aligned}$$

After the final weight matrix $\boldsymbol{w}^*$ is obtained, the selected features can be acquired with formula Eq. (3).

D. MODEL SAVING

In this paper, a special method is used to save the best performed model in training procession, which is considered to be an aspect of the performance improvement of the model. The model was usually trained with either early stopping strategy or optimal retention strategy. The early stop strategy can effectively reduce the training time, but setting the tolerance to the early stop is a big issue. If the tolerance is too small, the model may not be trained to optimum, and too large will waste a lot of computing resources and time. In practice, usually only one of the accuracy or loss metrics is considered.

In order to obtain the good performance for both accuracy and loss metrics, we proposes a statistical optimal retention strategy. The strategy focuses on the loss optimizations, but allows 2.5% to float up and down, which is considered as an insignificant change. If the model trained by the epoch has a better rate of accuracy appearing within the range of fluctuations, the model will be saved since the loss optimality is a constant at this point. If the loss drops significantly, the current model is considered to be the optimal model. Otherwise, the current model will not be saved because its performance has not been improved.

The method used in this paper is a combination of early-stop strategy and statistical optimal retention strategy, which not only retains the advantages of the early-stop strategy but also preserves optimal model.

E. ENSEMBLE METHOD

The cross-validation is often used in the training process of the BCI system to improve the generalization performance. The cross-validation usually divides the training set into K equal parts, one of which is used as the validation set, and the rests are used as the training set to train the model. This is also called K-fold cross-validation. When evaluating the model's performance, the average performance of each cross-validation model on the test set is taken as the performance of this model on the data set. The cross-validation method will reduce the size of the training set and the validation set, and it is suitable for the large-sized and medium-sized data sets. For the small-sized data sets, the model training and verification will be insufficient, especially the neural network models are difficult to train. Actually, the training data size is usually small in MI BCI feild, so it is not suitable to directly use the cross-validation method. Therefore, this paper adopts the average-validation method and simplified bagging method to improve the performance of the model.

The average-validation method is to perform K independent training on the same training set and test set. It requires random initialization for each independent model, and the feeding order of the training data is also random. That is to say, each fold is trained from a different starting point, and the trajectory of gradient descent is different as well. These ensure that the models will not converge to the same local minimum with a high probability, and will converge to the global minimum (if it exists), which means it has a certain generalization performance.

The bagging method is a classical application of ensemble learning. It generally generates multiple weak classifiers through data sampling. Whether we use the cross-validation or the average validation, K models will be eventually produced in the validation procedure. The performance of these K models is different, and none of them are very strong classifiers especially for insensitive subjects. If they can play a role in forecasting together, the better results can be achieved. So this paper uses a simplified bagging method, that is called stacking, to integrate them, thereby enhancing the performance of the final model.

We let $c_1, \ldots, c_k$ be the classifiers, $1 \leq k \leq K$, $\boldsymbol{x}$ be the training trials and $\boldsymbol{y}$ be the groudtruth. The simplified bagging method can be expressed as finding the optimal weights $\alpha_1, \ldots, \alpha_k$ so that $\left(\alpha_1 c_1(\boldsymbol{x}) + \cdots + \alpha_k c_k(\boldsymbol{x})\right) - \boldsymbol{y}$ is the smallest. That is to solve the following optimal problem as Eq. (9) shown:

$$\begin{aligned} &\arg \min_{\alpha_1,\ldots,\alpha_k} \left(\alpha_1 c_1(\boldsymbol{x}) + \cdots + \alpha_k c_k(\boldsymbol{x})\right) - \boldsymbol{y}, \\ &s.t.\ \ \alpha_1 + \cdots + \alpha_k = 1, \\ &\qquad \alpha_1, \ldots, \alpha_k \geq 0. \end{aligned} \tag{9}$$

Let $A = [\alpha_1, \alpha_2, \ldots, \alpha_k]$, and let $C(\boldsymbol{x}) = [c_1(\boldsymbol{x}), c_2(\boldsymbol{x}), \ldots, c_k(\boldsymbol{x})]^T$. Eq. (9) can be rewritten as the Linear Regression (LR) problem shown in Eq. (10):

$$\arg \min_{A} \frac{1}{2} \|AC(\boldsymbol{x}) - \boldsymbol{y}\|_2^2 \tag{10}$$

Since the weight matrix A has found, it can use Eq. (11) to classify trials.

$$Class = AC(\boldsymbol{x}) \tag{11}$$

IV. RESULTS & VISUALIZATION

A. RESULTS

Although we hope to present a complete end-to-end model in this paper, the regularization coefficient still needs to be specified artificially. In order to avoid the influence of subjective priori information on the model, the grid search method is used to select the regularization coefficients L_1, $\text{L}_{2,1}$ and TL_1 automatically. L_1 is the coefficient of Sparse Lasso, and $\text{L}_{2,1}$ is the coefficient of the Group Lasso, and TL_1 is the coefficient of the Time Constrain penalty. Each grid search training uses 5-fold average-validation. The results on BCI Competition IV 2a dataset are shown in Table 2.

TABLE 2. Selection of regularization parameters for different subjects on BCI Competition IV 2a dataset.

Subject	L_1	$\text{L}_{2,1}$	TL_1
01	2.5×10^{-5}	2.5×10^{-5}	7.5×10^{-6}
02	1.0×10^{-3}	1.0×10^{-4}	7.5×10^{-6}
03	1.0×10^{-4}	7.5×10^{-5}	2.5×10^{-6}
04	7.5×10^{-5}	1.0×10^{-4}	1.0×10^{-5}
05	2.5×10^{-5}	1.0×10^{-4}	7.5×10^{-6}
06	5.0×10^{-5}	1.0×10^{-4}	1.0×10^{-6}
07	7.5×10^{-5}	1.0×10^{-4}	2.5×10^{-6}
08	1.0×10^{-3}	1.0×10^{-4}	5.0×10^{-6}
09	7.5×10^{-5}	1.0×10^{-4}	2.5×10^{-5}

The 5-fold average-validation training using the regularization coefficients selected by grid search is compared with other baseline models. It should be noted that the TSGL-EEGNet and the EEGNet used the saving-model method are proposed in this paper. The results are shown in Table 3. It can be seen that our method reaches 81.34% average accuracy and 0.7511 average kappa, which is obviously higher than other methods. When comparing with the EEGNet, it can be seen that the TSGL-EEGNet has effectively reduced the over-fitting and improved the accuracy. When comparing with other neural network methods such as the MB3DCNN, it can be seen that the TSGL-EEGNet usually has more advantage in kappa than accuracy, which explains that the TSGL-EEGNet performs well in all classes. When comparing with other traditional methods such as the SS-MEMDBF, it can be seen that the TSGL-EEGNet has more advantage on insensitive subjects, which explains that the TSGL-EEGNet is more suitable for most people. As shown in Table 4, the two-sided p-values of the proposed method and the others are all less than 0.05. The T Test shows the two results do not have the same mean and the KS Test shows the two results are differently distributed. This shows that the method proposed in this paper is a significant improvement on the former, which is more effective and robust.

To fully prove the effectiveness of TSGL-EEGNet, we use an additional dataset BCI Competiton III IIIa to test the proposed method. The Grid Search results are shown in Table 5. The 5-fold average-validation results are shown in Table 6. We don't calculate the p-value on this dataset, because the number of subjects in this dataset is too small that the p-value is not statistical. It can be seen that 'TSGL-EEGNet (16,10) stacking' reaches 88.89% average accuracy and 0.8519 average kappa, which higher than $1st$ method by 7.48% and higher than the EEGNet by 11.04% according to kappa. This can show that the proposed method could be widely used.

B. VISUALIZATION

All of the following model-related images are determined by the specific data and models. This does not mean that other data and models will have the same images, but some common knowledge can be gained from it. The best model is from Subject 03, which has the highest decoding accuracy, and here we use it as the visualization model. In feature selection, both the EEGNet and the proposed model use 1-D convolution, so the frequency feature of the model may be decided by both of the 1-D convolution of feature extraction and selection section. It adds the unnecessary complexity to interpret the whole model. Therefore, in this part, the 1-D convolution and its variant in feature selection are replaced by 1 × 1 convolution. This can effectively avoid the interference of other factors, and show more clearly what the model learned, and explain the role of TSGL regularization.

We produce the Fourier Transform on the results for each frequency filter to find the certain frequency features that the model learned. As shown in Fig. 4, the horizontal axis is the frequency, and the vertical axis is the amplitude. Different colors of lines represent the different channels, and different frequency-filters are represented as different letters. From this we can see that the filters are obtained from different frequency information, and different channels (electrodes)

TABLE 3. Accuracy (kappa) of different models on the dataset BCI Competition IV 2a. The TSGL-EEGNet and EEGNet are using 5-fold average-validation, and the others are from references.

Subject	TSGL-EEGNet (16, 10)	TSGL-EEGNet (16, 10) stacking	EEGNet (16, 10)	SS-MEMDBF [26]	MB3DCNN [7]	C2CM [5]	TSSM+ SVM [27]	FBCSP [28]	FBCSP [23]
01	83.77 (0.7836)	85.41 (0.8054)	84.52 (0.7936)	**(0.86)**	77.40 (0.699)	**87.50** (0.833)	(0.77)	76.00	(0.68)
02	70.18 (0.6023)	**70.67(0.6091)**	61.17 (0.4821)	(0.24)	60.14 (0.459)	65.28 (0.537)	(0.33)	56.50	(0.42)
03	94.36 (0.9248)	95.24 (0.9365)	**95.90(0.9453)**	(0.70)	82.93 (0.788)	90.28 (0.870)	(0.77)	81.25	(0.75)
04	75.88 (0.6777)	**80.26(0.7361)**	66.58 (0.5532)	(0.68)	72.29 (0.594)	66.67 (0.556)	(0.51)	61.00	(0.48)
05	64.35 (0.5249)	70.29 (0.6044)	60.22 (0.4695)	(0.36)	**75.84(0.647)**	62.50 (0.500)	(0.35)	55.00	(0.40)
06	65.67 (0.5420)	68.37 **(0.5779)**	56.65 (0.4217)	(0.34)	**68.99** (0.538)	45.49 (0.273)	(0.36)	45.25	(0.27)
07	88.95 (0.8528)	**90.97(0.8797)**	85.78 (0.8104)	(0.66)	76.04 (0.653)	89.58 (0.861)	(0.71)	82.75	(0.77)
08	83.84 (0.7845)	**86.35(0.8180)**	84.83 (0.7978)	(0.75)	76.86 (0.702)	83.33 (0.778)	(0.72)	81.75	(0.75)
09	83.64 (0.7818)	84.47 (0.7929)	78.90 (0.7185)	**(0.82)**	**84.67** (0.713)	79.51 (0.727)	(0.83)	70.75	(0.61)
Ave.	78.96 (0.7194)	**81.34(0.7511)**	74.95 (0.6658)	(0.60)	75.02 (0.644)	74.46 (0.659)	(0.594)	67.75	(0.57)

TABLE 4. Accuracy (kappa)'s p-value of the purposed method 'TSGL-EEGNet (16,10) stacking' to other methods on the dataset BCI Competition IV 2a.

Test	TSGL-EEGNet (16, 10)	EEGNet (16, 10)	SS-MEMDBF [26]	MB3DCNN [7]	C2CM [5]	TSSM+ SVM [27]	FBCSP [28]	FBCSP [23]
T test	0.004 (0.004)	0.006 (0.006)	(0.014)	0.024 (0.003)	0.024 (0.023)	(0.002)	< 0.001(< 0.001)	
KS test	< 0.001 (< 0.001)	< 0.001 (< 0.001)	(0.006)	0.034 (< 0.001)	< 0.001 (< 0.001)	(< 0.001)	< 0.001(< 0.001)	

TABLE 5. Selection of regularization parameters for different subjects on BCI Competition III IIIa dataset.

Subject	L_1	$L_{2,1}$	TL_1
$K3$	5.0×10^{-3}	1.0×10^{-4}	7.5×10^{-4}
$K6$	1.0×10^{-4}	7.5×10^{-4}	2.5×10^{-5}
$L1$	7.5×10^{-4}	1.0×10^{-3}	7.5×10^{-4}

are similar in one filter. Some filters have similar effects, such as Filter (e), (f) and (g), but some filters are not doing anything, such as Filter (l) and (p).

Fig. 5 averages the frequency features in Fig. 4 for each class. As shown in Fig. 5, it can be seen that the frequency bands mainly cover $0-30$ Hz and $38-55$ Hz. According to [6], [18], we divide the frequency into 5 bands, where $2-8$ Hz is δ and θ rhythm, and $8-12$ Hz is α rhythm, and $12-20$ Hz is low β rhythm, and $20-30$ Hz is high β rhythm, and > 30 Hz is γ rhythm. Therefore, the frequency features learned by the model cover δ, θ, α, low β, high β and γ rhythm. What's more, there is not much difference in the frequency features of each class. It shows that the unselected features can't use directly to classify EEG signals.

The similarities of the depth-wise convolution and the CSP spatial filtering have been illustrated in Section III-B. To conveniently observe the depth-wise convolution, the spatial topology map is shown in Fig. 6, in which it uses the numbers to represent these spatial filters and uses the same letters in Fig. 4 to represent frequency filters. In our method, the data is standardized that the mean is 0 and the standard deviation is 1. Additionally, the distribution of the weights should be the same as the data, that is, the mean is 0 and the standard deviation is 1, and some weights could be negative values. If two spatial patterns are complementary, they are somewhat equivalent since they maybe work together for a signal which has both positive and negative values. It can be found in Fig. 6 that after passing through the frequency filter of similar frequency band, the spatial filter is also similar such as filter (b), (g), (k), and there are some symmetrical filters in each group of spatial filters such as filter $(b, 7)$ and $(b, 8)$, $(k, 3)$ and $(k, 4)$. It also can be seen that there are many spatial filters that pick up features near C3, C4,Cpz, and Pz, which are common spatial features that distinguish left and right hand imagery.

In feature selection, the visualized model uses 1×1 convolution and TSGL regularization penalty to select and fuse the features. The inputs of this part are all the features, and the outputs are the selected and fused features. In this paper, the selected and fused features are called new features, while the unselected features are called original features. These new features are the direct basis for classification, so they represent what the model really learns and what is really useful in the data. If the regulation of the new features is reflecting by using Grad-CAM [30] algorithm, the features that contribute to classification can be obtained. This paper presents an overview of the features of each class, as shown in Fig. 7 and Fig. 8.

It can be seen that the selected new features are clearly distinguished among the different classes. In terms of frequency

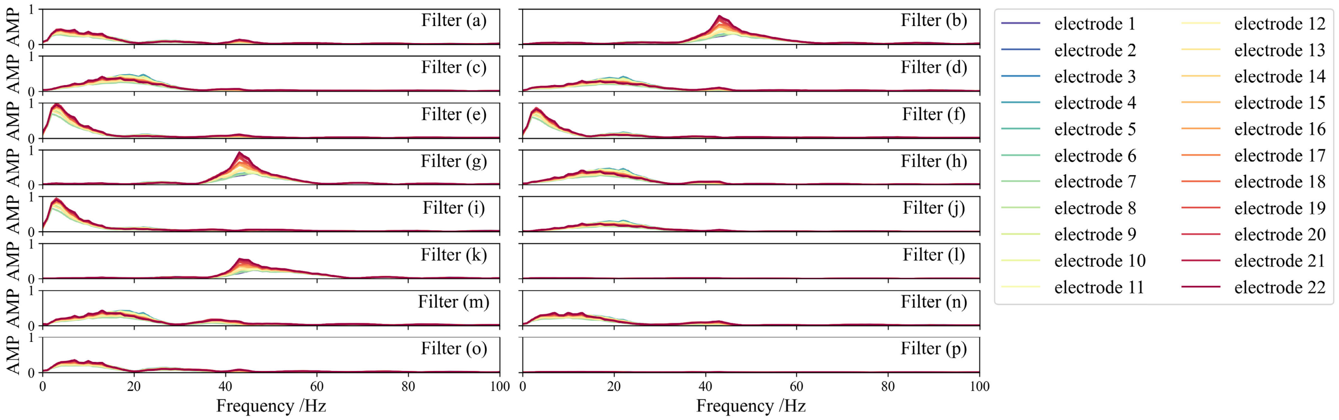

FIGURE 4. TSGL-EEGNet original frequency features on Subject 03 for each frequency filter.

TABLE 6. Accuracy (kappa) of different models on the dataset BCI Competition III IIIa. The TSGL-EEGNet and EEGNet are using 5-fold average-validation, and 1st and 2nd are the winners of the competition. (1*st*: Fisher ratios of channel-freqency-time bins, feature selection, designing mu and beta passband, multiclass CSP, SVM; 2*nd*: surface laplacian, 8-30Hz filter, CSP (one-vs-rest), SVM+kNN+LDA, bagging).

Subject	TSGL-EEGNet $(16, 10)$	TSGL-EEGNet $(16, 10)$ stacking	EEGNet $(16, 10)$	MFTFS [29]	$1st$	$2nd$
$K3$	$94.22\ (0.9230)$	$\mathbf{96.67(0.9556)}$	$95.11\ (0.9348)$	$79\ (0.71)$	(0.8222)	(0.9037)
$K6$	$76.00\ (0.6800)$	$80.83\ (0.7444)$	$73.00\ (0.6400)$	$\mathbf{84(0.77)}$	(0.7556)	(0.4333)
$L1$	$85.67\ (0.8089)$	$\mathbf{89.17(0.8556)}$	$78.17\ (0.7089)$	$89\ (0.85)$	(0.8000)	(0.7111)
Ave.	$85.30\ (0.8040)$	$\mathbf{88.89(0.8519)}$	$82.09\ (0.7672)$	$84\ (0.78)$	(0.7926)	(0.5222)

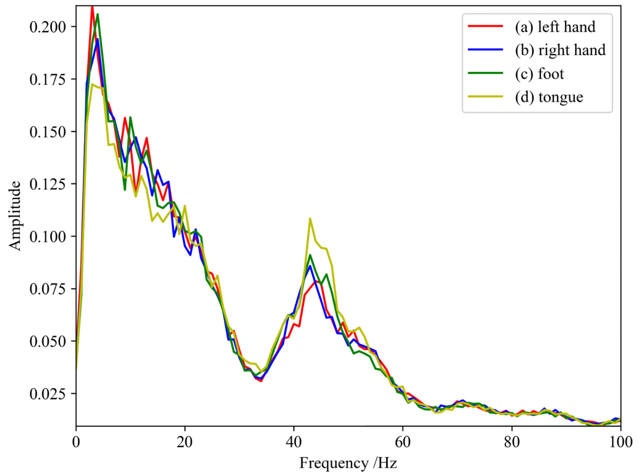

FIGURE 5. TSGL-EEGNet original frequency features on Subject 03 for each class.

features, the left hand (Fig. 7(a)) and the foot (Fig. 7(c)) have relatively high amplitude features, and the right hand (Fig. 7(b)) and the foot (Fig. 7(d)) features have low amplitude. It mainly show that the rhythm of $\delta, \theta, \alpha, \beta$ of the right hand is inhibited to the left hand at 2-25 Hz. The foot (Fig. 7(c)) is similar to the left hand, which implies that spatial features may be different. The difference between the features of the foot and tongue (Fig. 7(d)) is similar to that between the left and right hands. Moreover, there are some differences between the right hand and the tongue, mainly reflected in the inhibition of the δ, θ, α rhythm of the right hand motor imagery, which is the new knowledge for MI tasks. In terms of the spatial features, as shown in Fig. 8, the model obviously has leaned the different features for each class and the interesting-band. From Fig. 8(a) and Fig. 8(b), it can be known that the model exactly picks up features near C3 and C4 electrodes for right and left hand motor imagery, and it is well matched ERD and ERS in α and β bands. The left hand (Fig. 8(a)) extracts the features of the right hemispheres in α band and the left and right hemispheres in β band, and the right hand (Fig. 8(b)) is the opposite of the left hand. When we consider it combining with the frequency features, it is consistent with the known facts that the amplitude of the contralateral sensorimotor cortex signal increases and the contralateral β band signal decreases simultaneously. Fig. 8(c) reveals that the spatial feature of the foot motor imagery is concentrated on electrode Cpz. Fig. 8(d) is similar to Fig. 8(a), but the tongue movement's δ, θ, α rhythm amplitude are lower than left hand. Additionally, it can be found that 38-55 Hz γ band frequency features amplitude for all classes are almost the same and the spatial teatures are the same too, which suggests γ band signals may be noises.

Compared with the EEGNet (Fig. 9 and Fig. 10), the proposed method has some similarities with EEGNet. But the EEGNet's features has little difference among classes

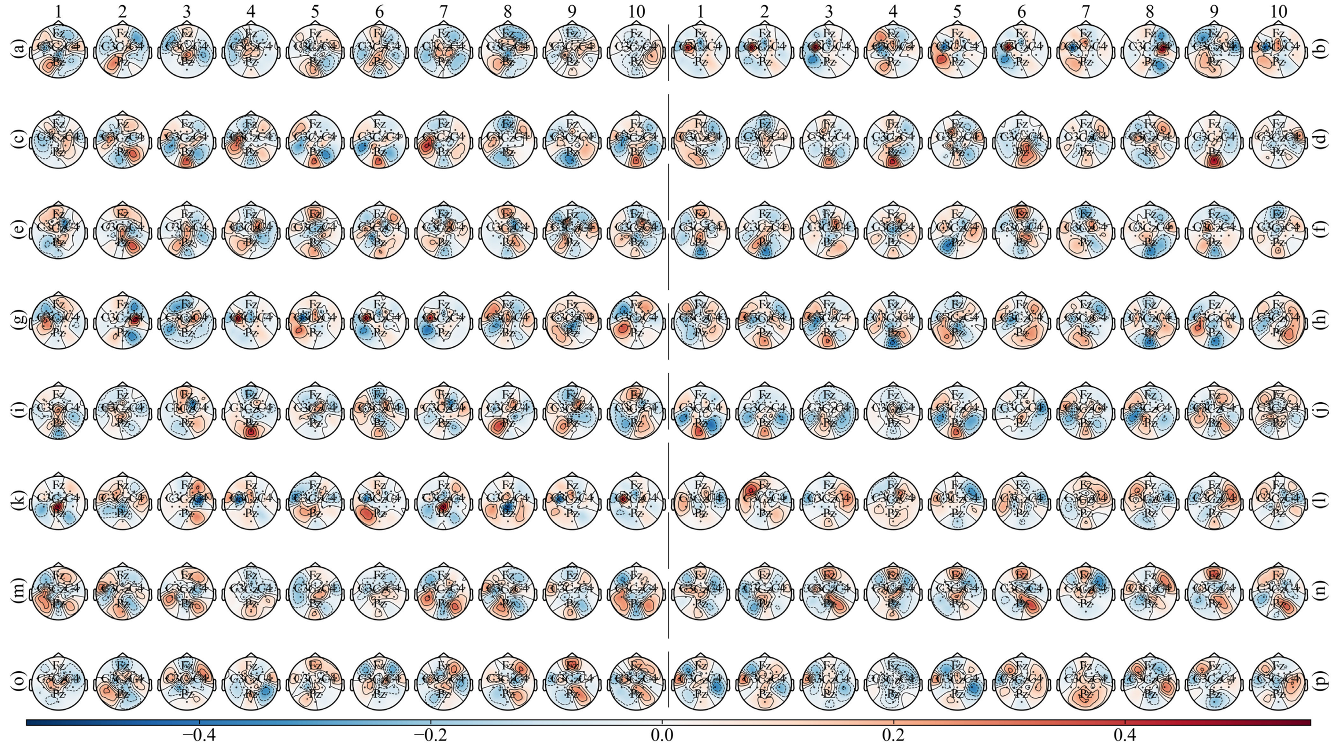

FIGURE 6. **Topomaps of TSGL-EEGNet depth-wise convolution weights.**

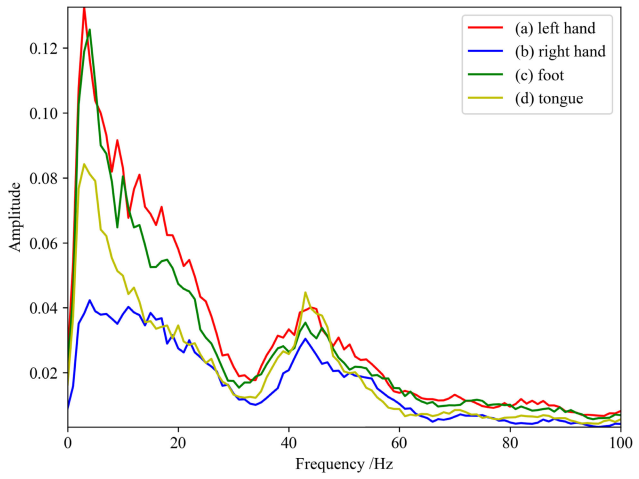

FIGURE 7. **TSGL-EEGNet selected frequency features on Subject 03 for each class.**

and interesting-bands. It can be found that the EEGNet seems to pay more attention to γ band features than the TSGL-EEGNet from Fig. 9 and Fig. 10. Additionally, the EEGNet don't have significantly ERD and ERS from Fig. 10(a) and Fig. 10(b). These may be the performances of over-fitting. The proposed method can learn more different features of classes, and can effectively avoid the over-fitting, which is an important reason why this method is better than the EEGNet.

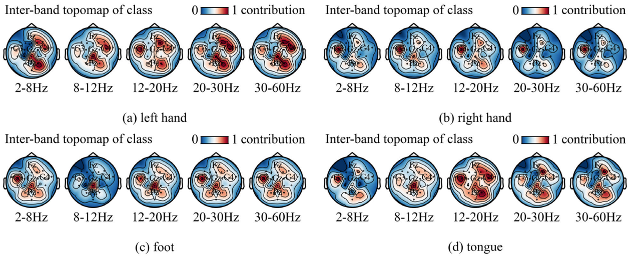

FIGURE 8. **TSGL-EEGNet selected spatial features in interesting-band on Subject 03 for each class.**

In addition to the above results, this paper also does some research to try to explain why different subjects have different accuracy rates, even uses the same best model. This issue will be discussed in detail in the discussion section.

V. DISCUSSION

A. THE PROBLEM OF PERSON SENSITIVITY

To analyze this problem of the person sensitivity, this paper chooses another Subject, Subject 06, whose model has the worst decoding accuracy. We compares his optimal model with the optimal model in section IV-B to find out the feature difference between them. As shown in Fig. 11 and Fig. 12, we can find that the EEGNet is difficult to learn enough useful information from the subject data with low decoding accuracy. There is no significant difference in frequency features

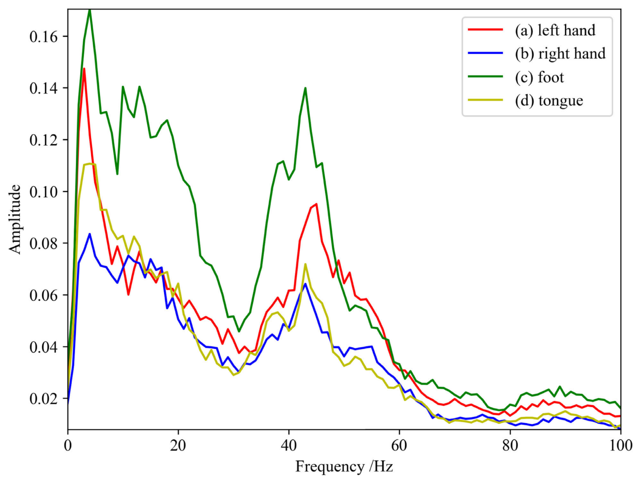

FIGURE 9. EEGNet selected frequency features on Subject 03 for each class.

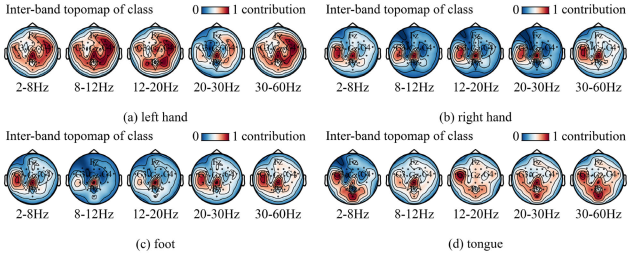

FIGURE 10. EEGNet selected spatial features in interesting-band on Subject 03 for each class.

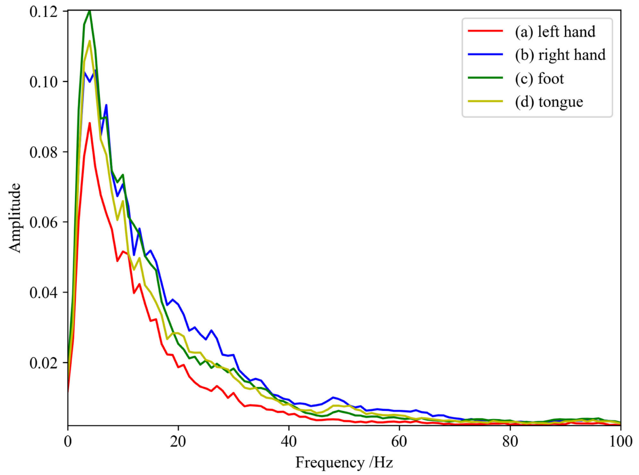

FIGURE 11. EEGNet selected frequency features on Subject 06 for each class.

among the four classes, and the spatial features repeat the same pattern in all interesting-bands of one class.

By using the TSGL-EEGNet, the result is better, but is still not enough. As shown in Fig. 13 and Fig. 14, we can find that the four classes have some obvious differences in frequency features as well as the spatial features. This may be the reason why the performance of the TSGL-EEGNet is better than the EEGNet. But these features are not as remarkable as the features of Subject 03, as shown in Fig. 7 and Fig. 8. This is probably the reason why the models don't work for everyone, and it is a common phenomenon that may be related to some specific factors such as skull thickness, density, shape, and so on. How to adapt the model to all people, or how to develop a targeted model to help these people, will be a long-term problem to be solved.

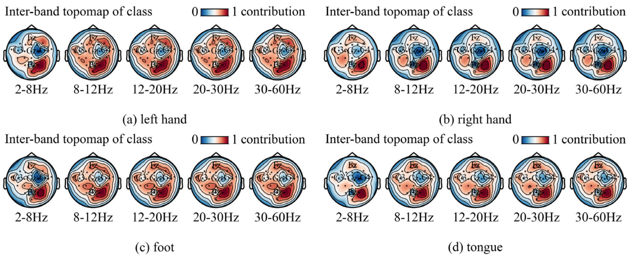

FIGURE 12. EEGNet selected spatial features in interesting-band on Subject 06 for each class.

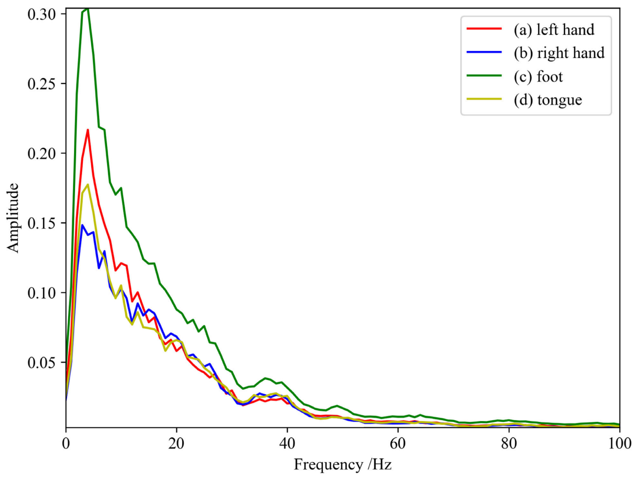

FIGURE 13. TSGL-EEGNet selected frequency features on Subject 06 for each class.

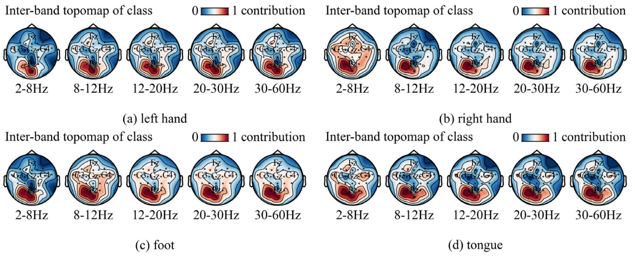

FIGURE 14. TSGL-EEGNet selected spatial features in interesting-band on Subject 06 for each class.

B. SELECTING THE OPTIMAL TIME-SEGMENT

Since the EEG signals are time-varying and non-stationary, it is important to select the optimal time-segment. Generally, there are three kinds of methods to select time windows, experience-based [13], [23], [26], [31], search-based [32], and data-based. The experience-based method is obviously not good, and the search-based method requires training a huge number of models, so both are not suitable for the neural networks. However, the data-based methods almost depend on the data, which would serve as the best choice for neural networks. Additionally, the cropped training also would be a good choice to select the time-segment when using a point-to-point model. The cropped training is usually used in the popular researches referred to [6], [7], and it is a data reinforcement method during training period. It crops a long data into many intersecting short ones which share the same label, which likes using a window to slide throughout the long data. The data size can be increased greatly using this method, so that the size of the EEG data set is no longer small. For example, the dataset used in this paper has a 4-second motor imagery time and is sampled at a rate of 250 Hz, so each sample has 1000 data points. Since the effective rhythm of motor imagery is usually greater than 4 Hz (α, β, γ rhythms), and according to the Nyquist-Shannon Sampling Theorem, this paper uses a time window of 2s (500 data points) to cut the data every 25 (< 31.25) data points. Thus, each sample can generate 21 data, which all share the same label. That is, the training and the test data size are both 21 times larger than before and the data length is reduced to 2s. During the training, the samples will be classified according to the different subjects, and then each subject's samples will be loaded in random order.

The cropped training method crops a long data into many intersecting short ones which share the same label. Thus, a new neural network structure for short data is obtained. As a fact, the main discrimination between online and offline methods is the time requirement for computing. Online methods emphasize immediacy which is difficult for offline methods. However, the time requirement can be cut down using the cropped training when a method can output the classification results in the confidence level. Thus, the cropped training can help the offline methods convert to the online methods. The classification confidence level is the variance computed by the outputs from the last Softmax activation layer. Obviously, the larger the variance is, the higher the confidence is presented. For N classes task, the largest variance is $\frac{N-1}{N^2}$ would be reached when outputs have only one element values 1, and others value 0. In this way, all the algorithms designed for off-line systems can be easily applied to on-line systems, which will be a direction of our future researches.

VI. CONCLUSION

In this work, we propose a neural network model TSGL-EEGNet, which has the good performance on the MI EEG and the well interpretability for itself. The TSGL-EEGNet is improved based on a popular deep learning model EEGNet, and uses the traditional machine learning algorithm for optimization. The proposed model in our work based on the EEGNet is consistent with the principle of CSP algorithm, which maximizes the variance of class features through learning a spatial filter. Based on the public datasets, the proposed method reaches the 81.34% average classification accuracy and the 0.7511 average classification kappa on the BCI Competition IV 2a dataset. Its 4-classes classification accuracy and kappa are significantly greater than EEGNet (74.95% 0.6658), SS-MEMDBF (0.60), MB3DCNN (75.02% 0.644), C2CM (74.46% 0.659) and FBCSP (67.75% 0.57). Additionally, the proposed method also reaches 88.89% average classification accuracy and 0.8519 average classification kappa on the BCI Competition III IIIa dataset, which is greater than the others as well. The results show that it is an effective way to improve the classification accuracy by merging the traditional machine learning and deep learning algorithms. Furthermore, the proposed model is somewhat interpretable. On one hand, after the mathematical explanation, it can be proved that the proposed deep learning model and CSP algorithms have equivalent feature extraction and selection parts. On the other hand, through the visualization, it can also be proved that the proposed model can learn meaningful features, such as the features of reflecting ERSs and ERDs in α and β bands. From the visualization, it can be seen that the deep learning models can have the same interpretability as some traditional machine learning algorithms. Thus, the interpretability of the deep learning models could be achieved by incorporating with some traditional mathematical algorithms. But this is not to say that we should be content with the traditional algorithms. On the contrary, the deep learning has the advantages that some traditional algorithms cannot achieve, such as end-to-end models, adaptive hyperparameters learning, and high classification accuracy. This suggests that we need to re-study in the traditional fields to get better, faster and more interpretable neural network models.

REFERENCES

[1] P. Bashivan, I. Rish, M. Yeasin, and N. Codella, "Learning representations from EEG with deep recurrent-convolutional neural networks," 2015, *arXiv:1511.06448*. [Online]. Available: http://arxiv.org/abs/1511.06448

[2] M.-A. Li, Y.-F. Wang, S.-M. Jia, Y.-J. Sun, and J.-F. Yang, "Decoding of motor imagery EEG based on brain source estimation," *Neurocomputing*, vol. 339, pp. 182–193, Apr. 2019.

[3] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, "EEGNet: A compact convolutional neural network for EEG-based brain–computer interfaces," *J. Neural Eng.*, vol. 15, no. 5, Oct. 2018, Art. no. 056013.

[4] N. Padfield, J. Zabalza, H. Zhao, V. Masero, and J. Ren, "EEG-based brain–computer interfaces using motor-imagery: Techniques and challenges," *Sensors*, vol. 19, no. 6, p. 1423, Mar. 2019.

[5] S. Sakhavi, C. Guan, and S. Yan, "Learning temporal information for brain–computer interface using convolutional neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 11, pp. 5619–5629, Nov. 2018.

[6] R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, and T. Ball, "Deep learning with convolutional neural networks for EEG decoding and visualization," *Hum. Brain Mapping*, vol. 38, no. 11, pp. 5391–5420, Nov. 2017.

[7] X. Zhao, H. Zhang, G. Zhu, F. You, S. Kuang, and L. Sun, "A multi-branch 3D convolutional neural network for EEG-based motor imagery classification," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 10, pp. 2164–2177, Oct. 2019.

[8] K.-W. Ha and J.-W. Jeong, "Motor imagery EEG classification using capsule networks," *Sensors*, vol. 19, no. 13, p. 2854, Jun. 2019.

[9] B. Graimann, B. Allison, and G. Pfurtscheller, "Brain–computer interfaces: A gentle introduction," in *Brain-Computer Interfaces*. Berlin, Germany: Springer, 2009, pp. 1–27.

[10] L. F. Nicolas-Alonso and J. Gomez-Gil, "Brain computer interfaces, a review," *Sensors*, vol. 12, no. 2, pp. 1211–1279, 2012.

[11] G. Pfurtscheller and A. Aranibar, "Event-related cortical desynchronization detected by power measurements of scalp EEG," *Electroencephalogr. Clin. Neurophysiol.*, vol. 42, no. 6, pp. 817–826, Jun. 1977.

[12] G. Pfurtscheller, C. Brunner, A. Schlögl, and F. H. L. da Silva, "Mu rhythm (de)synchronization and EEG single-trial classification of different motor imagery tasks," *NeuroImage*, vol. 31, no. 1, pp. 153–159, May 2006.

[13] X. Deng, D. Li, J. Mi, F. Gao, Q. Chen, J. Wang, and R. Liu, "Motor imagery ECoG signal classification using sparse representation with elastic net constraint," in *Proc. IEEE 7th Data Driven Control Learn. Syst. Conf. (DDCLS)*, May 2018, pp. 44–49.

[14] G. Rodríguez-Bermúdez and P. J. García-Laencina, "Automatic and adaptive classification of electroencephalographic signals for brain computer interfaces," *J. Med. Syst.*, vol. 36, no. S1, pp. 51–63, Nov. 2012.

[15] A. Schlögl, F. Lee, H. Bischof, and G. Pfurtscheller, "Characterization of four-class motor imagery EEG data for the BCI-competition 2005," *J. Neural Eng.*, vol. 2, no. 4, pp. L14–L22, Dec. 2005.

[16] J. Zhou, M. Meng, Y. Gao, Y. Ma, and Q. Zhang, "Classification of motor imagery eeg using wavelet envelope analysis and LSTM networks," in *Proc. Chin. Control Decis. Conf. (CCDC)*, Jun. 2018, pp. 5600–5605.

[17] Y. Hashimoto and J. Ushiba, "EEG-based classification of imaginary left and right foot movements using beta rebound," *Clin. Neurophysiol.*, vol. 124, no. 11, pp. 2153–2160, Nov. 2013.

[18] K. K. Ang, Z. Y. Chin, H. Zhang, and C. Guan, "Filter bank common spatial pattern (FBCSP) in brain–computer interface," in *Proc. IEEE Int. Joint Conf. Neural Netw., IEEE World Congr. Comput. Intell.*, Jun. 2008, pp. 2390–2397.

[19] Y. Zhang, C. S. Nam, G. Zhou, J. Jin, X. Wang, and A. Cichocki, "Temporally constrained sparse group spatial patterns for motor imagery BCI," *IEEE Trans. Cybern.*, vol. 49, no. 9, pp. 3322–3332, Sep. 2019.

[20] Y. Jiao, Y. Zhang, X. Chen, E. Yin, J. Jin, X. Wang, and A. Cichocki, "Sparse group representation model for motor imagery EEG classification," *IEEE J. Biomed. Health Informat.*, vol. 23, no. 2, pp. 631–641, Mar. 2019.

[21] S. Scardapane, D. Comminiello, A. Hussain, and A. Uncini, "Group sparse regularization for deep neural networks," *Neurocomputing*, vol. 241, pp. 81–89, Jun. 2017.

[22] C. Brunner, R. Leeb, G. Müller-Putz, A. Schlögl, and G. Pfurtscheller, "BCI competition 2008–graz data set A," Inst. Knowl. Discovery, Lab. Brain-Comput. Interfaces, Graz Univ. Technol., Graz, Austria, 2008, pp. 136–142, vol. 16. [Online]. Available: http://www.bbci.de/competition/iv/desc_2a.pdf

[23] K. K. Ang, Z. Y. Chin, C. Wang, C. Guan, and H. Zhang, "Filter bank common spatial pattern algorithm on BCI competition IV datasets 2a and 2b," *Frontiers Neurosci.*, vol. 6, p. 39, Mar. 2012.

[24] J. Jin, Y. Miao, I. Daly, C. Zuo, D. Hu, and A. Cichocki, "Correlation-based channel selection and regularized feature optimization for MI-based BCI," *Neural Netw.*, vol. 118, pp. 262–270, Oct. 2019.

[25] J. Jin, R. Xiao, I. Daly, Y. Miao, X. Wang, and A. Cichocki, "Internal feature selection method of CSP based on L1-norm and Dempster–Shafer theory," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Aug. 24, 2020, doi: 10.1109/TNNLS.2020.3015505.

[26] P. Gaur, R. B. Pachori, H. Wang, and G. Prasad, "A multi-class EEG-based BCI classification using multivariate empirical mode decomposition based filtering and Riemannian geometry," *Expert Syst. Appl.*, vol. 95, pp. 201–211, Apr. 2018.

[27] X. Xie, Z. L. Yu, H. Lu, Z. Gu, and Y. Li, "Motor imagery classification based on bilinear sub-manifold learning of symmetric positive-definite matrices," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 6, pp. 504–516, Jun. 2017.

[28] H. Bashashati, R. K. Ward, and A. Bashashati, "User-customized brain computer interfaces using Bayesian optimization," *J. Neural Eng.*, vol. 13, no. 2, Apr. 2016, Art. no. 026001.

[29] Y. Yang, S. Chevallier, J. Wiart, and I. Bloch, "Subject-specific time-frequency selection for multi-class motor imagery-based BCIs using few Laplacian EEG channels," *Biomed. Signal Process. Control*, vol. 38, pp. 302–311, Sep. 2017.

[30] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," *Int. J. Comput. Vis.*, vol. 128, no. 2, pp. 336–359, Oct. 2019.

[31] S. Taran, V. Bajaj, D. Sharma, S. Siuly, and A. Sengur, "Features based on analytic IMF for classifying motor imagery EEG signals in BCI applications," *Measurement*, vol. 116, pp. 68–76, Feb. 2018.

[32] J. Feng, E. Yin, J. Jin, R. Saab, I. Daly, X. Wang, D. Hu, and A. Cichocki, "Towards correlation-based time window selection method for motor imagery BCIs," *Neural Netw.*, vol. 102, pp. 87–95, Jun. 2018.

XIN DENG (Member, IEEE) received the bachelor's degree from the Department of Computer Science and Technology, Jilin University, Changchun, China, in 2004, the master's degree from the Department of Computer Science, Chongqing University, Chongqing, China, in 2007, and the Ph.D. degree in computer engineering from the National University of Singapore, Singapore, in 2013. He is currently an Associate Professor with the College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, China. His research interests include brain–computer interface, electrophysiological signal processing, data engineering, and machine learning.

BOXIAN ZHANG received the B.Sc. degree in mathematics, in 2018. He is currently pursuing the master's degree with the Chongqing University of Posts and Telecommunications, Chongqing, China. His research interests include brain–computer interface, brain science, machine learning, and artificial intelligence.

NIAN YU received the Ph.D. degree in geophysics from the School of Geophysics, Chengdu University of Technology, in 2012. He currently works as an Associate Professor with the School of Electrical Engineering, Chongqing University. His main research interests include electrical signal processing, electromagnetic field simulation and applications, and artificial intelligence.

KE LIU received the B.S. degree in automatic control from Southwest University, Chongqing, China, in 2011, and the Ph.D. degree in pattern recognition and intelligent systems from the South China University of Technology, Guangzhou, China, in 2016. He is currently an Associate Professor with the College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing. His research interests include pattern recognition and Bayesian inference and their applications in EEG data analysis.

KAIWEI SUN received the bachelor's degree in information security and the master's degree in computer technology from the Chongqing University of Posts and Telecommunications, China, in 2010 and 2013, respectively, and the Ph.D. degree in information and communication engineering from Inha University, South Korea, in 2017. He is currently an Associate Professor with the School of Computer Science and Technology, Chongqing University of Posts and Telecommunications. His research interests include machine learning, big data analysis, computer vision, and nature language processing.