Classification of Motor Imagery Based on Multi-Scale Feature Extraction and the Channel-Temporal Attention Module

Runze Wu, Jing Jin, *Senior Member, IEEE*, Ian Daly, Xingyu Wang, and Andrzej Cichocki, *Life Fellow, IEEE*

***Abstract*— Motor imagery (MI) is a popular paradigm for controlling electroencephalogram (EEG) based Brain-Computer Interface (BCI) systems. Many methods have been developed to attempt to accurately classify MI-related EEG activity. Recently, the development of deep learning has begun to draw increasing attention in the BCI research community because it does not need to use sophisticated signal preprocessing and can automatically extract features. In this paper, we propose a deep learning model for use in MI-based BCI systems. Our model makes use of a convolutional neural network based on a multi-scale and channel-temporal attention module (CTAM), which called MSCTANN. The multi-scale module is able to extract a large number of features, while the attention module includes both a channel attention module and a temporal attention module, which together allow the model to focus attention on the most important features extracted from the data. The multi-scale module and the attention module are connected by a residual module, which avoids the degradation of the network. Our network model is built from these three core modules, which combine to improve the recognition ability of the network for EEG signals. Our experimental results on three datasets (BCI competition IV 2a, III IIIa and IV 1) show that our proposed method has better performance than other state-of-the-art methods, with accuracy rates of 80.6%, 83.56% and 79.84%. Our model has stable performance in decoding EEG signals and achieves efficient classification performance while using fewer network parameters than other comparable state-of-the-art methods.**

***Index Terms*— Motor imagery, EEG, multi-scale convolution, convolution neural network, attention module.**

Manuscript received 31 March 2023; revised 9 June 2023; accepted 1 July 2023. Date of publication 12 July 2023; date of current version 2 August 2023. This work was supported in part by the Scientific and Technological Innovation (STI) 2030-Major Projects under Grant 2022ZD0208900; in part by the National Natural Science Foundation of China under Grant 62176090; in part by the Shanghai Municipal Science and Technology Major Project under Grant 2021SHZDZX; in part by the Program of Introducing Talents of Discipline to Universities through the 111 Project under Grant B17017; in part by the Shuguang Project supported by the Shanghai Municipal Education Commission and the Shanghai Education Development Foundation under Grant 19SG25; in part by the Polish National Science Center under Grant UMO-2016/20/W/NZ4/00354; in part by the National Government Guided Special Funds for Local Science and Technology Development (Shenzhen, China) under Grant 2021Szvup043; and in part by the Project of Jiangsu Province Science and Technology Plan Special Fund in 2022 (Key Research and Development Plan Industry Foresight and Key Core Technologies) under Grant BE2022064-1. *(Corresponding author: Jing Jin.)*

Runze Wu and Xingyu Wang are with the Key Laboratory of Smart Manufacturing in Energy Chemical Process, Ministry of Education, Shanghai 200237, China (e-mail: epoch_666@163.com; xywang@ecust.edu.cn).

Jing Jin is with the Key Laboratory of Smart Manufacturing in Energy Chemical Process, Ministry of Education, Shanghai 200237, China, and also with the Shenzhen Research Institute, East China University of Science and Technology, Shenzhen 518063, China (e-mail: jinjingat@gmail.com).

Ian Daly is with the Brain-Computer Interfacing and Neural Engineering Laboratory, School of Computer Science and Electronic Engineering, University of Essex, Colchester, CO4 3SQ Essex, U.K. (e-mail: i.daly@essex.ac.uk).

Andrzej Cichocki is with the Laboratory for Advanced Brain Signal Processing, RIKEN Brain Science Institute, Wako 351-0198, Japan, also with the Systems Research Institute, Polish Academy of Sciences, 01-447 Warsaw, Poland, and also with the Department of Informatics, Nicolaus Copernicus University, 87-100 Torun, Poland (e-mail: a.cichocki@riken.jp).

Digital Object Identifier 10.1109/TNSRE.2023.3294815

I. Introduction

BRAIN-COMPUTER Interface (BCI) uses artificial intelligence to decode signals from the brain in order to provide a communication pathway between the brain and the world [1]. The original purpose of BCI technology is to identify the intention of human activities by analyzing EEG signals and converting them into commands to control external auxiliary devices, to assist people with motor disabilities to interact with the external environment [2]. As a result of continuous development of research, BCI technology is gradually being applied to increasing numbers of fields such as the medical industry [3], [4], entertainment [5], smart homes [6], and the military [7].

Commonly used BCI paradigms include steady-state visual evoked potentials (SSVEPs), P300 potentials, and motor imagery. Among these paradigms, motor imagery requires no additional stimulation apparatus, instead users modulate their EEG simply through the imagination of movement. In contrast to other BCI systems, the motor imagery paradigm can directly map a user's movement intention to an action. This allows participants to complete specific tasks by imagining limb movements. Furthermore, BCIs based on motor imagery can be spontaneous. In other words, participants can generate EEG signals through motor imagery without the need for external cues or other stimuli. Consequently, motor imagery BCIs have become one of the most popular paradigms.

When imagining movement, the activity of specific frequency bands within the brain changes. Specifically, activity in the mu band, from 8-12 Hz, and the beta band, from 13-30 Hz, are known to change during motor imagery. While performing an imagined task, such as the right-hand movement, the contralateral hemisphere of the brain exhibits a phenomenon of reduced low-frequency activity, termed event-related desynchronization (ERD), while the ipsilateral hemisphere of the brain produces a phenomenon of increased activity, termed the event-related synchronization (ERS) [8].

To accurately recognize the ERD/S, considerable research efforts have focused on combinations of feature extraction methods combined with machine learning to complete the classification task. Within the feature extraction step a form of transformation method (usually a linear transformation) is employed to identify and extract some important features from the EEG signals. The important feature information is retained and the influence of noisy additional information is reduced or removed, thus transforming the originally complex high-dimensional EEG signals into a lower-dimensional less noisy domain [9]. Feature extraction is usually conducted from four perspectives: the time domain [10], [11], the frequency domain [12], [13], the time-frequency domain [14], [15], and the spatial domain [16], [17]. Typical feature extraction methods include, but are not limited to, the Fast Fourier Transform (FFT) [18], the wavelet transform (WT) [19], principal component analysis (PCA) [20], and common spatial patterns (CSP) [21]. Among these methods, the CSP algorithm is the most widely used in BCI systems. Its basic principle is to identify a spatial filter that maximizes the variance between two categories. Recently, the basic CSP algorithm has been extended in a number of ways to meet particular challenges within the BCI domain. Among these extensions, the filter bank common space mode (FBCSP) [22] is one promising method that uses the frequency domain characteristics of MI to optimize the spatial filter. Specifically, within the FBCSP algorithm, the MI signal is divided into multiple frequency sub-bands and then each sub-band is filtered by CSP to extract an optimal feature set.

The feature extraction step is typically followed by feature classification. The classification task makes use of machine learning methods to identify user intention from the extracted features. Common machine learning methods applied within the BCI field include, but are not limited to, support vector machines (SVMs) [23], [24], linear discriminant analysis classifiers (LDAs) [25], [26], K-nearest neighbours classifiers (KNNs) [27], [28], and naive Bayes classifiers (NBs) [29], [30].

However, traditional machine learning methods work most effectively only when the features they are applied to have been carefully chosen and pre-processed to maximize the signal-to-noise ratio. An alternative approach, that many investigators have recently begun to focus on, is deep learning. Deep learning models can effectively capture a high-dimensional feature representation from the EEG signals as well as the potential relationships between internal features through a nonlinear deep structure. For EEG signals with complex information content and strong time-varying characteristics, a deep feature representation can be extracted through deep learning models. Deep learning models do not require complex processing of the input data, and some models can even directly use the original data as their input without any pre-processing or feature extraction [31]. Many different deep learning models have been developed over recent years to classify EEG data. For example, Liu and Zeng [32] proposed a multi-feature fusion method based on ResNet to extract features. This model classified EEG with accuracies which were 39.65% higher than those achieved by a model using single features. For feature extraction and classification of MI-EEG signals, Wang et al. [33] combined a Squeeze-and-Excitation convolution neural network (SECNN) with the time-varying autoregressive model (TVAR) and power spectral density (PSD) time-frequency analysis, to improve the accuracy of MI-BCI systems. Hwaidi and Chen [34] trained deep neural networks by combining a deep autoencoder (DAE) and CNN architectures to classify EEG MI signals. The results of the model show that it outperforms current CNN-based approaches and several traditional machine-learning approaches.

Deep learning does not require complex feature extraction methods, but simple data preprocessing steps have been shown to further improve their classification results. For example, Shalu et al. [35] achieved relatively good results by transforming continuous wavelet transforms into time-frequency plots and feeding them into a deep CNN for classification. Zhu et al. [36] designed a separate channel convolution network to encode the multi-channel data of CSP, preserving the time-varying information that is helpful for distinguishing tasks. However, some studies have proved that deep learning can even directly extract features from raw EEG data. For example, Wu et al. [37] proposed a parallel multi-scale filter bank CNN for MI classification. The extracted output features are connected to the spatial convolution layer to complete the fusion of multimodal features from EEG signals. Roy et al. [38] explored the use of different fusion models to automatically complete multimodal feature extraction and classification from raw EEG signals. The model achieves 80.32% accuracy on the BCI competition IV 2b dataset. Li et al. [39] used amplitude interference as a means of data enhancement to expand the dataset and constructed a channel projection mixed-scale CNN to decode EEG. This framework used the original multi-channel EEG signal as the input, and achieved an average accuracy of 67.17% in a four-class classification task.

Deep learning has considerable potential to improve the performance of MI-EEG BCIs, but some problems still remain: 1) Most methods developed to date are intended for binary classification tasks, or if they are applied to multi-class problems, convert the multi-class problem into multiple binary tasks. However, there is a growing need for effective multi-class solutions for MI BCIs. 2) The process of feature extraction and selection is very time-consuming. Ideally, we would like to take full use of the advantages of automatic learning of deep neural networks. To do this we need to extract more effective features by either adopting data enhancement

methods or by designing more complex network models. However, complex networks will increase the run time and the number of parameters that need to be trained in the network. 3) There are individual differences in the EEG signals of different participants but the single-scale convolution kernel can only use a single set of weights when extracting features. 4) Although the neural network can automatically extract features, the extracted features are not necessarily effective in all cases. Extracting features without emphasis will not only increase the calculation cost but also lead to feature redundancy.

The literatures on MI mentioned above are all based on the use of 2D inputs to the network, while there are far fewer studies on the use of 1D inputs. Liu et al. [40] compared the classification results of both one-dimensional (1D) and two-dimensional (2D) input forms based on public datasets and the results indicate that the 1D form of input can lead to higher classification accuracies and converge faster. Jia et al. [41] used a 1D input, but the convolution was performed in the time dimension and finally reached an average classification accuracy rate of 78% on the four classified published datasets, demonstrating that a 1D input can also be helpful for MI classification tasks.

To tackle the problems listed above, a deep learning-based multi-class MI signal recognition method is proposed in this paper. This model utilizes preprocessed EEG signals to realize end-to-end automatic learning without the need for manually designed feature extraction methods. The main contributions of this study are as follows:

1) To investigate multi-class tasks, this paper carries out experiments on two BCI competition datasets, each containing four-classes of movement tasks. To demonstrate the performance of the model we add a dataset with 2-categories of MI tasks.

2) To improve the efficiency of feature extraction, this paper proposes an end-to-end neural network and uses the proposed data augmentation to enrich the feature information.

3) In view of the differences in EEG signals recorded from different participants, a multi-scale module is designed to extract richer features, which increases the range of the network to learn features and improves the classification accuracy.

4) To address the problem that the neural network may learn features that are not focused, the information over different channels and over time is learned through two modules, a channel attention module and a temporal attention module, to attempt to improve the classification result.

The rest of the paper is organized as follows: the details of the methods are described in Section II. The experiments and results are presented in Section III. The factors influencing the experimental and future work are discussed in Section IV. Finally, we conclude our research in Section V.

II. Methods

The model proposed in this paper is a neural network with a multi-scale module and two attention modules. The overall framework of the MSCTANN model is shown in Figure 1. The model includes three core parts: the multi-scale module, the residual module and the channel-temporal attention module (CTAM). Augmentation of the training data is used to provide more information for the neural network. Multi-scale modules then automatically extract features from this augmented data and different extraction levels solve the problem of differences in EEG signals among participants. The residual module is used to fuse the features transmitted by the multi-scale module, and the introduction of the residual module avoids network degradation caused by an excessive number of network layers. The CTAM is used to automatically select the fused features, effectively avoiding information redundancy, and automatically learning the importance of different features, thus improving the classification result for MI-EEG signals.

A. Data Augmentation

For neural networks, the data that need to be used for training needs to be sufficiently larger. However, most EEG datasets cannot satisfy this requirement to support training the network. Therefore, data enhancement is necessary. Each sample of EEG data based on MI in this paper is represented as a 2D matrix of dimensions $C \times T$ (channel $\times$ time), where rows represent data collected from different electrodes and columns represent data at different sample points. In this paper, a head-to-tail extended data augmentation method is proposed. The schematic diagram of the head-to-tail augmentation method is shown in Figure 1(a). For each trial, the signal head is extracted at a certain length and filled into the tail. This extraction process is continuously cycled until the cycle of the whole signal is completed. The length of the loop is an adjustable parameter and one of the influential factors affecting the final classification outcome. If the length of the loop is too long, the network cannot obtain enough information to overcome the over-fitting problem. Conversely, the difference between different samples will be very small.

B. The Multi-Scale Module

There are several challenges still to be overcome when designing MI-BCIs. One of the most important of these challenges is the difference in EEG signals between different participants. As a result of this, if a single extraction method is used, it will not only limit the extraction of each participant's information but also ignore the individual differences between different participants, resulting in a poor final classification result. How to use the information provided to extract more features is a problem that still needs a solution. Therefore, this paper designs a multi-scale structure, which automatically extracts features from the original EEG signals based on multi-scale convolution and pooling. Its structure is shown in Figure 1(b).

The multi-scale structure proposed in this paper is designed according to related methods in the field of signal processing. Conv1 is a convolutional layer with a smaller kernel, which can effectively collect fine-grained local information. Conv2 is a convolutional layer with a medium kernel, which can retain relatively coarse-grained feature information. Conv3 is a convolutional layer with a larger kernel, which can capture the overall characteristics of the EEG signals. Three different sizes

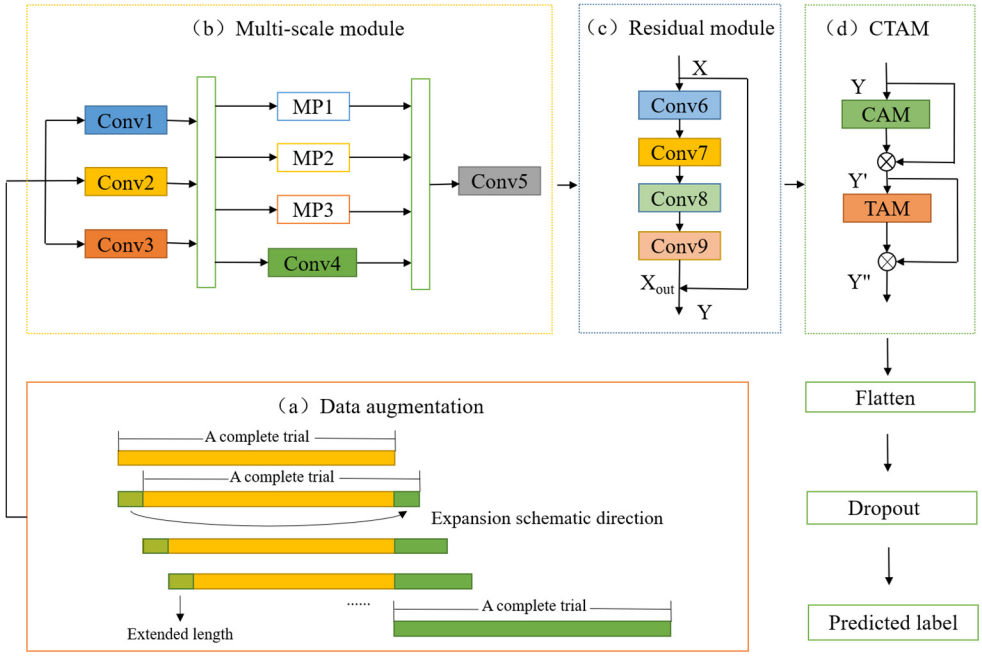

Fig. 1. The model architecture and the training process, based on multi-scale feature extraction.

of convolutional kernels can extract more adequate features from a multi-scale perspective by the multi-scale structure. In order to extract features it is necessary to reduce the matrix parameters and feature dimensions through the pooling layer, thereby reducing the number of parameters in the last fully connected layer. The incorporation of the pooling layer can also speed up calculations and prevent overfitting effects. Most of the existing studies used a single-scale pooling layer, which increased the possibility of information loss to some extent. Although this method can remove redundant information, the criteria for information redundancy are not fixed for different participants. If only a single scale is used for extraction and processing, it greatly increases the possibility of loss of important information. Therefore, our method added multi-scale pooling to multi-scale convolution, and two multi-scale structures were combined to better process the MI-EEG signals.

C. The Residual Module

The residual module can fuse the extracted features. Furthermore, the introduction of this module avoids the problem of network degradation produced by an excessive number of network layers. The structure of the residual module is shown in Figure 1(c). The connection line on the right side of the module is called the identity shortcut connection, which adds neither extra parameters nor computational complexity. The identity shortcut connection can solve the problem that the newly added layer does not work effectively by allowing the module to skip one or more layers if needed. This module is formed by combining multiple 1D convolutional layers and batch normalization (BN) layers with the superimposed residual connection. The definition of this module can be expressed as:

$$Y = X_{out} + X \tag{1}$$

where X and X_{out} represent the input and the output of the residual module, and Y represents the total output. The features learned by the shallow network can be passed to the deep network by the residual connectivity module, thus avoiding network degradation.

D. Channel-Temporal Attention Module (CTAM)

Convolutional block attention module (CBAM) is a lightweight attention module that can conduct attention training in both channel and spatial dimensions [42]. Inspired by the CBAM module, this article constructs a channel-temporal attention module, called CTAM. The overall structure of our CTAM module is shown in Figure 1(d), and the specific structure is shown in Figure 2. The CTAM module includes a channel attention module and a temporal attention module, which complement both channel attention and temporal attention, achieving considerable performance improvement while keeping the computational overhead small. Further directed screening of features can be performed to automatically learn more important features, thus achieving the goal of boosting the MI-EEG signal classification performance.

1) Channel Attention Module (CAM): The channel attention module compresses the temporal dimension without changing the channel dimension. This module focuses on useful information of the target. The specific flow of the channel attention module is shown in Figure 2(a). It utilizes two parallel max pooling layers and average pooling layers for feature selection and compression. Then the number of channels is compressed to 1 channel per reduction fold of the original by sharing the MLP module. After the Relu activation function gets the result of two activations, the channel number is expanded to the original number. These two output results are added element by element to get the output result of the channel attention via a sigmoid activation function. The formula for channel

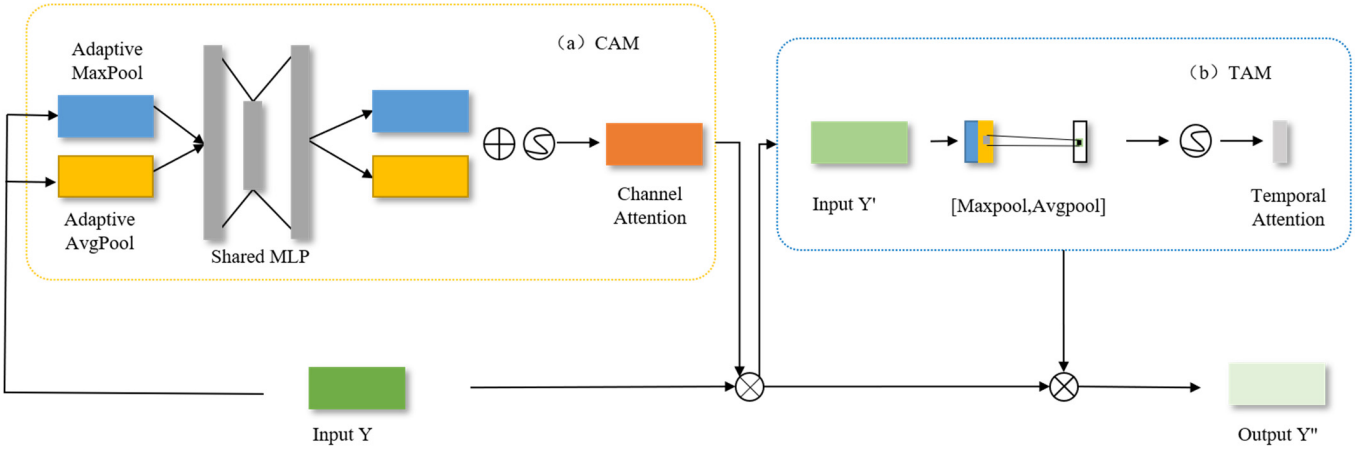

Fig. 2. The framework structure of the CTAM module adapted in our proposed model.

attention is as follows:

$$M_c(Y) = \sigma\left(MLP\left(AvgPool\left(Y\right)\right) + MLP\left(MaxPool\left(Y\right)\right)\right) = \sigma\left(W_1\left(W_0\left(Y^c_{avg}\right)\right) + W_1\left(W_0\left(Y^c_{max}\right)\right)\right) \quad (2)$$

where $M_c(Y)$ represents the convolution result of the CAM, σ represents the activation function. W_1 and W_0 represent the weights of the MLP and Y^c_{avg} and Y^c_{max} represent features output by different pooling layers under the channel attention module.

2) Temporal Attention Module (TAM): TAM compresses the channel dimension without changing the temporal dimension. This module focuses on the location information of the target. The specific flow of the temporal attention module is shown in Figure 2(b).

The TAM produces two feature maps by maximum pooling and average pooling of the output from the channel attention module. Then the two feature maps are combined and turned into a single-channel feature map by the convolution operation. The feature map of temporal attention is obtained through the sigmoid function. Finally, the output is multiplied by the original map. The formula for temporal attention is as follows:

$$M_s(Y) = \sigma\left(f\left(\left[AvgPool\left(Y\right); MaxPool\left(Y\right)\right]\right)\right) = \sigma\left(f\left(\left[Y^s_{avg}; Y^s_{max}\right]\right)\right) \quad (3)$$

where $M_s(Y)$ represents the convolution result of the TAM, f represents the active convolutions, and Y^s_{avg} and Y^s_{max} represent the output features of the different pooling layers under the temporal attention module.

3) Combined Channel and Temporal Attention Module: Attention modules can increase the representativeness of the network by focusing on important features, and suppressing unnecessary ones. The CTAM module emphasizes temporal information while simultaneously reinforcing channel information.

The CTAM module consists of the CAM and TAM modules. Through the CAM module, the input feature Y multiplies the result by the original input, and the result Y' is the input of the TAM module. Finally, the output result of the TAM module is multiplied with Y':

$$Y' = M_c(Y) \otimes Y$$
$$Y'' = M_s(Y') \otimes Y \quad (4)$$

where Y is the original input feature, Y' is the result of multiplying the CAM convolution output with the original map, and Y'' is the result after multiplying the TAM convolution output with Y'. This is also the CTAM final output result.

III. Experiments and Results

A. EEG Data

This paper uses three well-regarded datasets in the field of MI classification: Dataset 1: BCI competition IV 2a dataset [43], Dataset 2: BCI competition III IIIa dataset [44] and Dataset 3: BCI competition IV 1 dataset [43].

Dataset 1: The BCI competition IV 2a dataset contains data from nine participants performing four classes of MI tasks (involving the left hand, right hand, foot, and tongue). This dataset records EEG signals from an EEG setup placed according to the international 10-20 system with 25 electrodes (22 EEG channels and 3 EOG channels). The sampling frequency is 250 Hz and a 0.5-100 Hz band-pass filter and 50 Hz power frequency notch filter are used for filtering. Each participant completed two sessions, each of which contained six runs with 48 trials per run, while one session contained 288 trials.

The timing pattern of Dataset 1 is shown in Figure 3(a). In the experiment, the beginning of the trial is a fixation cross for the first 1s. Then a cue of direction shows for 1.25s. At t=3s, participants are asked to imagine the corresponding movement until they finished the task at t=6s. The acquisition of this dataset uses a feedback-free experimental paradigm, intercepting the time of the signal from 0.5s after the start of the cue to the end of MI, that is, from 2.5-6s, with a total intercept of 3.5s. Since MI is most commonly associated with changes in the α (8-12 Hz) and β (13-30 Hz) frequency bands the data is band-pass filtered at 8-30 Hz using a Butterworth filter.

Dataset 2: The BCI competition III IIIa dataset contains data from 3 participants performing a four-category MI task. The task type is the same as used in Dataset 1. This dataset

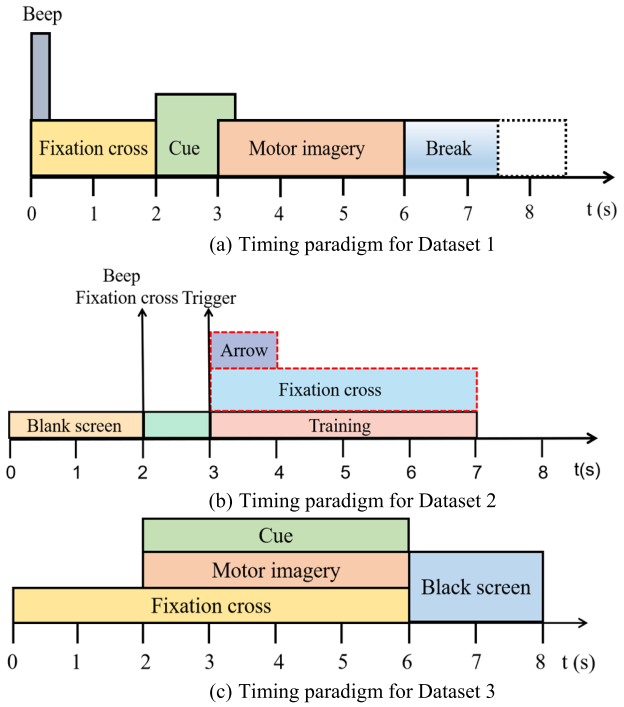

Fig. 3. Timing paradigm used during recording of the datasets.

includes EEG signals recorded with 60 electrodes. The sampling frequency used was 250 Hz. Participant K3 in this dataset completed 360 MI trials while the other two participants each completed 240 trials. The sample data is equal for each category.

The timing paradigm used to record Dataset 2 is shown in Figure 3(b). In the experiment, the beginning of the trial is a black screen for the first 2s. Then a fixation cross "+" is displayed for 1s. A directional arrow is then displayed for 1s. At the same time, the participant is asked to imagine the corresponding movement until the fixation cross disappears at t=7s. Each of the 4 cues is displayed 10 times within each run in a randomized order. The trials for Dataset 2 are extracted from 3.5-6.5s, and the filter settings used remain consistent with Dataset 1.

Dataset 3: The BCI competition IV 1 dataset contains data from 7 participants performing a 2-category MI task. This dataset includes EEG signals recorded with 59 electrodes. The sampling frequency used was 100 Hz. The data from participants labeled c, d, and e from this dataset were not used, because they are artificially generated. Each participant was asked to complete 200 trials.

The timing paradigm used to record Dataset 3 is shown in Figure 3(c). In the experiment, the beginning of the trial is a fixation cross for the first 2s. Then a directional arrow is displayed for 4s. At the same time, the participant is asked to imagine the corresponding movement until the cross disappears at t=6s. The trials for Dataset 3 are extracted from 2.5-5.5s, and the filter settings used remain consistent with Dataset 1.

B. Experimental Setup

This experiment adopts within-subject classification. We employ a fivefold cross-validation approach to perform network training by optimizing the cross-entropy loss function, with the number of training iterations set to 50. The neural network is trained using the Adam optimizer, which updates the network weights more efficiently than the classical random gradient descent method. Additionally, it also accelerates the convergence of the neural network. The initial learning rate of the network is set to 1×10.3 and then adjusted through a cosine annealing attenuation strategy, which means that the learning rate will be readjusted and restored after decay to a certain value, jumping out of the current local optimal solution and searching for the global optimal solution again. To prevent overfitting problems, a dropout rate of 0.4 is set in the final fully connected layer. More network parameter settings are detailed in Table I (note, the term N in the table denotes the batch size).

TABLE I
DETAILS OF THE CNN STRUCTURE USED WITH THE THREE DATASETS

Layer	Dataset1 Output Shape	Dataset1 Kernel Size	Dataset2 Output Shape	Dataset2 Kernel Size	Dataset3 Output Shape	Dataset3 Kernel Size
Conv1	[N,36,875]	3	[N,90,750]	7	[N,88,300]	3
Conv2	[N,36,875]	11	[N,90,750]	21	[N,88,300]	11
Conv3	[N,36,875]	19	[N,90,750]	35	[N,88,300]	19
Conv4	[N,54,875]	1	[N,135,750]	1	[N,132,300]	1
MP1	[N,54,875]	7	[N,135,750]	7	[N,132,300]	7
MP2	[N,54,875]	19	[N,135,750]	19	[N,132,300]	19
MP3	[N,54,875]	31	[N,135,750]	31	[N,132,300]	31
Conv5	[N,108,875]	1	[N,270,750]	1	[N,264,300]	1
Conv6	[N, 54, 875]	1	[N,135,750]	1	[N,132,300]	1
Conv7	[N,108,875]	3	[N,270,750]	3	[N,264,300]	3
Conv8	[N, 54, 875]	1	[N,135,750]	1	[N,132,300]	1
Conv9	[N,108,875]	3	[N,270,750]	3	[N,264,300]	3
CAM	[N, 108, 1]	-	[N, 270, 1]	-	[N,264,1]	-
SAM	[N, 1, 875]	7	[N, 1, 750]	7	[N,1,300]	7
CBAM	[N,108,875]	-	[N,270,750]	-	[N,264,300]	-
Flatten	108	-	270	-	264	-
Dropout	4	-	4	-	2	-

C. Overall Comparison

The model presented in this paper was compared with several classical and state-of-the-art models on the BCI IV 2a, BCI III IIIa and BCI IV 1 datasets. Table II compares the effect of our proposed method with other state-of-the-art methods on Dataset 1. The numbers highlighted in bold in the table indicate the participants' best outcomes.

We compared our model to the following methods:

1. FBCSP [45]: A model that manually extracts features. This model is often used as a baseline method to classify MI-EEG signals. It has yielded good results in several previous EEG decoding studies. It performs task classification by extracting CSP features from different frequency bands and then using the SVM model to classify the features.

2. EEGNet [46]: A deep learning model that uses 2D temporal convolution, deep convolution, and separable convolution to accomplish classification.

3. DeepConvNet [47]: A deep learning model that is deeper than the ShallowConvNet. It consists of four convolutional and Max pooling layer blocks, followed by a soft Max layer.

4. FBCNet [48]: A deep learning model using EEG bandpass filtering to create multi-frequency bands. It consists of two trainable layers.

TABLE II
A COMPARISON OF THE CLASSIFICATION PERFORMANCES ACHIEVED BY THE DIFFERENT MODELS ON DATASET 1 (THE *P*-VALUE IS THE RESULT OF A *t*-TEST COMPARING OUR PROPOSED METHOD TO EACH OF THE OTHER METHODS)

Method / Participant	FBCSP	EEGNet	Deep ConvNet	FBCNet	MBEEGSE	EEG-TCNet	Proposed
A01	76.74	84.72	81.25	84.07	86.11	87.84	**89.58**
A02	57.29	54.83	52.77	64.94	65.63	69.46	**73.62**
A03	85.06	88.17	85.75	88.55	**93.4**	90.97	93.04
A04	53.82	62.5	72.22	62.15	73.61	62.13	**75.33**
A05	54.88	68.33	68.33	60.82	**69.08**	67.38	64.24
A06	46.53	54.83	54.88	44.44	59.01	56.95	**68.06**
A07	85.75	86.43	68.08	86.44	**87.85**	85.41	87.17
A08	84.72	76.73	82.62	84.72	86.44	91.68	**90.30**
A09	82.29	75.42	82.28	82.62	79.84	77.76	**84.04**
Ave (%)	69.68	72.44	72.02	73.20	77.89	76.62	**80.60**
Sd	16.16	12.28	11.47	15.50	11.66	13.09	10.52
p-value	0.001	0.005	0.006	0.007	0.049	0.029	-

5. MBEEGSE [49]: A deep learning model for decoding MI known as a multi-branch EEGNet with squeeze-and-excitation blocks.

6. EEG - TCNet [50]: A deep learning model with temporal convolutional network. It involves dilated convolution and residual modules.

As shown in Table II, our proposed MSCTANN model can achieve an average recognition accuracy of 80.6%. Compared to other methods, our proposed MSCTANN method achieves a statistically significantly higher mean classification accuracy over participants. In terms of recognition accuracy, MSCTANN achieves an accuracy that is, on average, 10.92% higher than FBCSP, which proves that our model can extract more effective information than FBCSP. The five other deep-learning models, EEGNet, DeepConvNet, FBCNet, MBEEGSE and EEG-TCNet are generally more effective than FBCSP, which demonstrates the advantages of deep learning. However, our proposed MSCTANN method is, on average, 8.16%, 8.58%, 7.4%, 2.71% and 3.98% more accurate than these five models, which demonstrates the effectiveness of our proposed model design. The structure of the deep learning model is very important for feature extraction, and selecting an appropriate model can result in better classification performance. The reason our proposed MSCTANN model performs better than other state-of-the-art methods may be due to the use of multi-scale design elements and the role of the attention module. Specifically, our proposed MSCTANN model incorporates both a multi-scale model and attention modules, which makes feature extraction and screening more reasonable and leads to improved classification performance.

Figure 4 shows the results of our proposed MSCTANN model and the other state-of-the-art models on Dataset 2 and Dataset 3. In Dataset 2, our MSCTANN model achieves the highest accuracy, 83.56%. The performance of our MSCTANN model is 16.2% higher than FBCSP. When comparing the deep learning models, our MSCTANN model achieves an average accuracy that is 7.87%, 11.62%, 4.9%, 1.1% and 3.23% higher than those five models, indicating that targeted extraction of features can improve model performance. In Dataset 3, our MSCTANN model achieves the highest accuracy, 79.88%. The performance of our MSCTANN model is 17.13% higher than FBCSP. Compared to the five other deep learning models, our model produces a performance which is 8.75%, 13.88%, 6.13%, 0.88% and 2%, better respectively. When considering all three datasets, our model has a more stable performance than the other methods we compare against. In addition, the single training time for the three datasets is 64.98s, 65.8s, and 25.61s, respectively.

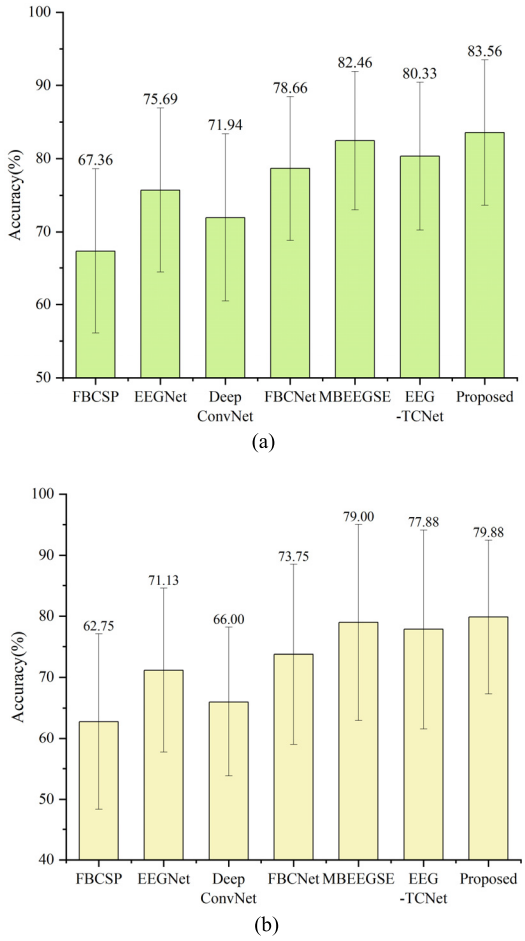

Fig. 4. A comparison of classification performances achieved by each model on (a) Dataset 2 and (b) Dataset 3.

IV. DISCUSSION

The performance of our proposed MSCTANN model is affected by several factors: 1) To demonstrate the advantages of multi-scale kernels, ablation experiments corresponding to single scale kernels are conducted. 2) In the multi-scale

module, the different sizes of the convolution kernels will affect the content of the information in the extracted features. 3) The validity of the CTAM layer for feature learning and selection. 4) The abundance of features as a result of the data augmentation and feature augmentation methods.

A. Influence of Multi-Scale Kernel

Multi-scale kernels can extract features at different scales at the same time. To highlight the advantages of multi-scale convolution kernels, we perform corresponding ablation experiments on the optimal combination of multi-scale kernels for each dataset. The results of the three datasets are shown in figures 5(a), (b) and (c), respectively. We use a radar chart to display the results. As can be seen from the figure, the single scale kernels of almost all participants are not as performant as multi-scale kernels. In dataset 1, compared to single scale kernels, the accuracy of multi-scale kernels for all participants improved by 5.57%, 5.27%, and 3.05%, respectively. In dataset 2, the accuracy improved by 2.82%, 6.94%, and 8.19%, respectively. In dataset 3, the accuracy improved by 4.25%, 2.13%, and 3.52%, respectively. This result also verifies that multi-scale kernel can extract more information than single scale kernel. In addition, we added a significance test for the single scale kernel (in Figure 5(d)). From a statistical point of view, the single scale kernels and multi-scale kernels have significant differences in results.

B. Multi-Scale Kernel Size

In this section, we use different combinations of convolution kernel sizes to explore the effect of kernel size on model performance. Due to the different number of channels that are available in the two datasets, kernel combinations need to be considered separately for each of the three datasets. The results for each dataset are shown in Figure 6. Due to the close number of EEG channels in dataset 2 and dataset 3, the same kernel combination is adopted for both these datasets. It can be seen that the most suitable kernel combination for Dataset 1 (in Figure 6(a)) is (3, 11, 19), the most suitable kernel combination for Dataset 2 (in Figure 6(b)) is (7, 21, 35) and the most suitable kernel combination for Dataset 3 (in Figure 6(c)) is (3, 11, 19). For the three datasets, the accuracy of the best kernel combination was 5.95%, 5.32%, and 3.04% higher than that of the worst kernel combination, respectively. From the experimental results achieved with the three datasets, it is not difficult to see that the performance difference between different kernel combinations is still relatively large. In this article, we were only able to consider a limited number of combinations, so there may still be better kernel combinations that we have not yet discovered. Indeed, from our current results, it is not yet possible to analyze helpful optimization rules and this, and related aspects, need to be further explored in the future.

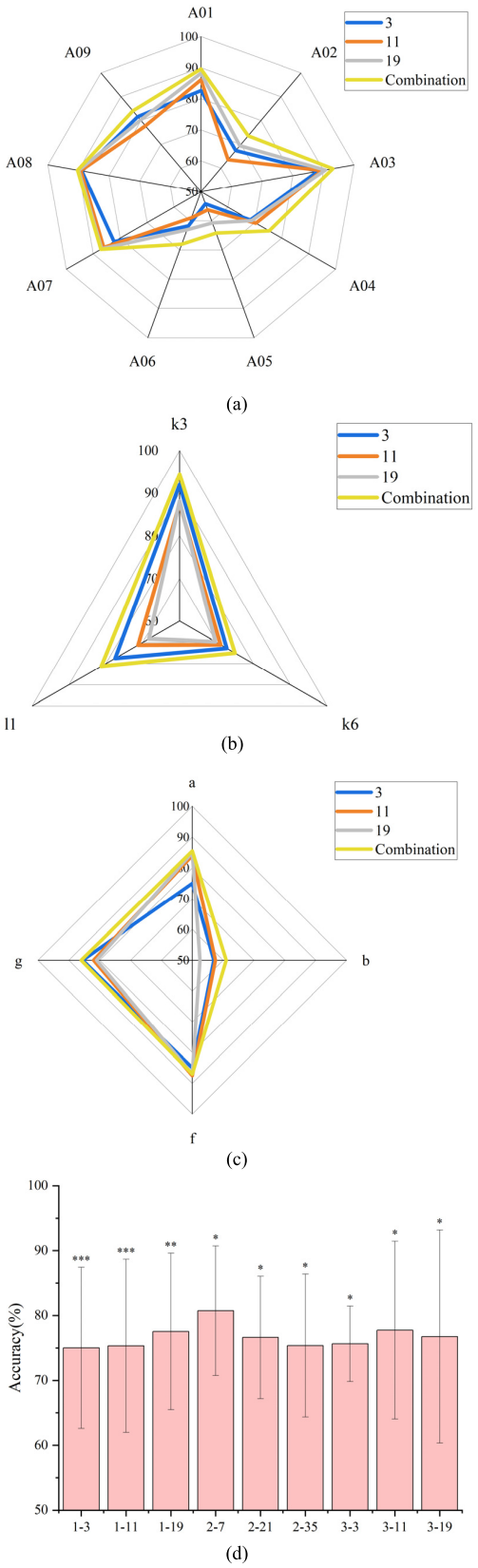

Fig. 5. Comparison of the performance between the optimal multi-scale kernel and the corresponding single scale kernel for each dataset. (a) Dataset 1 (b) Dataset 2 (c) Dataset 3 (d) Significance testing of single scale kernel for three datasets.

C. Influence of the CTAM

To explore the effects of the CTAM layer on the classification results, we perform ablation experiments. As shown in Figure 7. The experimental results on all datasets illustrate that the classification accuracy appears to be reduced by different amounts for each participant without the CTAM layer. If the participants with high and low accuracy rates are

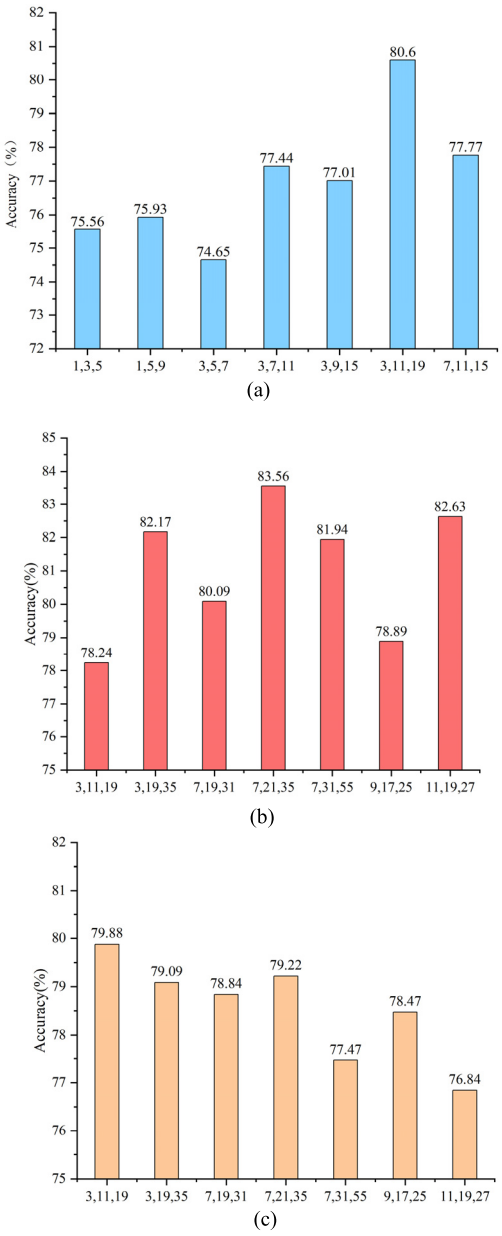

Fig. 6. Comparison of different classification performances achieved with each dataset with a range of multi-scale kernel sizes. (a) Performances achieved on Dataset 1 (b) Performances achieved on Dataset 2 (c) Performances achieved on Dataset 3.

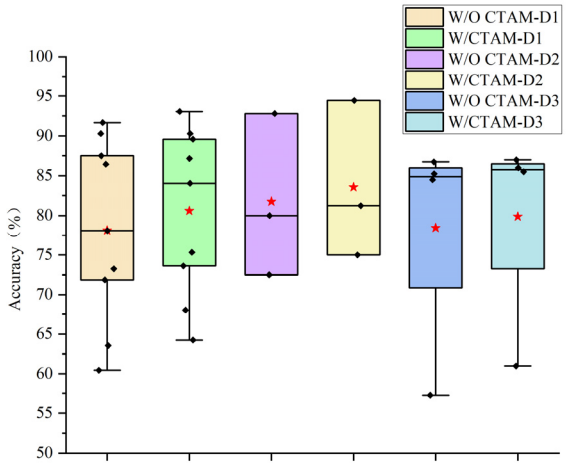

Fig. 7. Influence of the CTAM layer on the classification results. Black dots represent the accuracy of each participant. The red five-pointed stars represent the average values. D1 represents Dataset 1, D2 represents Dataset 2 and D3 represents Dataset 3.

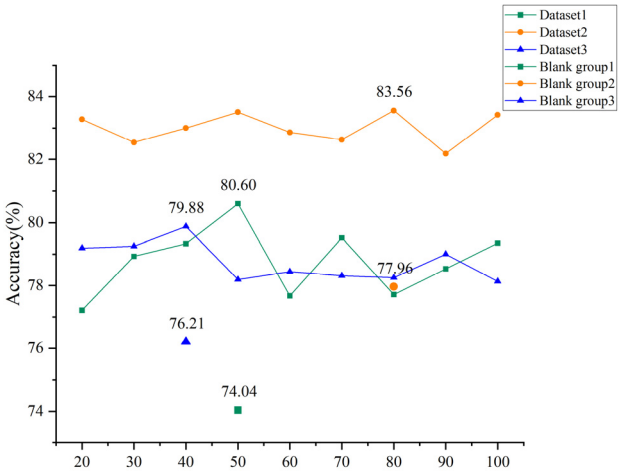

Fig. 8. Influence of data expansion length and multiples on the classification results. Blank groups represent the results of unused data augmentation.

divided by 78.89% (the average accuracy of all subjects), the results show that the effect of improvement of low accuracy rate participants is more obvious, with an average increase in performance of 3.45%, and the best improvement effect reaching 5.95%. Thus, the validity of the CTAM layer in our proposed model is demonstrated. The features extracted by the original EEG signals through the multi-scale module and the residual module are different. The CTAM layer can automatically learn the importance of different features, and then improve the classification result for the MI tasks.

D. Influence of Data Augmentation

The purpose of data augmentation is to optimize the training process by overcoming the problem of insufficient training data. To demonstrate the validity of our head-to-tail data augmentation method, the experiments are repeated with a dataset that does not use data augmentation. Figure 8 shows the test results achieved with this un-augmented data from each of the three datasets (the blank groups). The use of data augmentation has improved the classification performance of the three datasets by 7.99%, 6.58%, and 3.63%, respectively.

In the head-to-tail data augmentation method, the augmentation length determines the final amount of training data, which will, in turn, affect the training of the model. If the augmentation is too short, it may result in the reuse of data, which not only does not provide additional useful information but also increases the computational cost of training. If the augmentation is too long, it may again reduce the utilization of the data and prevent the extraction of additional useful information. To investigate the relationship between the expansion length and the final training effect, we conducted a comparison test at different amplification lengths. Considering computational cost and time, the expansion length is only set

from 20 to 100 in steps of 10. The results of this test on each of the two datasets are shown in Figure 8. In Dataset 1, the best effect is the combination with a length of 50. This results in a 3.39% improvement over the worst combination. In Dataset 2, the best effect is the combination with a length of 80. This results in an improvement of 1.38% over the worst combination. In Dataset 3, the best effect is the combination with a length of 40. This results in an improvement of 1.75% over the worst combination. This illustrates that there is not a clear relationship between the classification results and the augmentation length. A shorter augmentation length will obtain more training data, but it does not bring better results, and even requires more network computing power.

E. Future Work

In the future, we will investigate the application of lightweight networks for the classification of motor imagery, reducing the number of parameters in the neural networks and improving the operational efficiency of the neural networks. In addition, our future work will further investigate which network architectures are more suitable for processing MI-EEG signals and try to utilize fewer channels to achieve better results.

V. Conclusion

In this paper, we propose a neural network model called MSCTANN. It is a deep learning-based signal recognition method for multi-class MI classification. The multi-scale module of our MSCTANN model is able to automatically extract and screen features, which can extract rich feature information for the differences in MI-EEG signals. The CTAM layer in our proposed MSCTANN model is able to automatically learn channel and temporal valid information from the data, thus making the network more targeted for learning. Additionally, this paper also proposes a data augmentation method to increase the training data samples, which provides more information for our MSCTANN model. The validity of the method on two four-classification datasets is verified by experiments. Our MSCTANN model provides ideas for the model architecture of deep learning and makes contributions to the recognition task.

References

[1] J. Jin, Z. Wang, R. Xu, C. Liu, X. Wang, and A. Cichocki, "Robust similarity measurement based on a novel time filter for SSVEPs detection," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Oct. 14, 2021, doi: 10.1109/TNNLS.2021.3118468.

[2] X. Zhang, J. Jin, S. Li, X. Wang, and A. Cichocki, "Evaluation of color modulation in visual P300-speller using new stimulus patterns," *Cognit. Neurodynamics*, vol. 15, no. 5, pp. 873–886, Oct. 2021.

[3] M. Jochumsen, T. A. M. Janjua, J. C. Arceo, J. Lauber, E. S. Buessinger, and R. L. Kæseler, "Induction of neural plasticity using a low-cost open source brain-computer interface and a 3D-printed wrist exoskeleton," *Sensors*, vol. 21, no. 2, p. 572, Jan. 2021.

[4] N. Kobayashi and M. Nakagawa, "BCI-based control of electric wheelchair using fractal characteristics of EEG," *IEEJ Trans. Electr. Electron. Eng.*, vol. 13, no. 12, pp. 1795–1803, Dec. 2018.

[5] A. Kreilinger, H. Hiebel, and G. R. Muller-Putz, "Single versus multiple events error potential detection in a BCI-controlled car game with continuous and discrete feedback," *IEEE Trans. Biomed. Eng.*, vol. 63, no. 3, pp. 519–529, Mar. 2016.

[6] Q. Gao, X. Zhao, X. Yu, Y. Song, and Z. Wang, "Controlling of smart home system based on brain-computer interface," *Technol. Health Care*, vol. 26, no. 5, pp. 769–783, Oct. 2018.

[7] C. N. Munyon, "Neuroethics of non-primary brain computer interface: Focus on potential military applications," *Frontiers Neurosci.*, vol. 12, p. 696, Oct. 2018.

[8] W. Liang, J. Jin, I. Daly, H. Sun, X. Wang, and A. Cichocki, "Novel channel selection model based on graph convolutional network for motor imagery," *Cognit. Neurodynamics*, Oct. 2022, doi: 10.1007/s11571-022-09892-1.

[9] D. Coyle, T. M. McGinnity, and G. Prasad, "Improving the separability of multiple EEG features for a BCI by neural-time-series-prediction-preprocessing," *Biomed. Signal Process. Control*, vol. 5, no. 3, pp. 196–204, Jul. 2010.

[10] S. S. Gupta and R. R. Manthalkar, "Detection of motor activity in visual cognitive task using autoregressive modelling and deep recurrent network," in *Pattern Recognition and Data Analysis With Applications*. Singapore: Springer, 2022, pp. 371–381.

[11] M. Z. A. Faiz and A. A. Al-Hamadani, "Online brain computer interface based five classes EEG to control humanoid robotic hand," in *Proc. 42nd Int. Conf. Telecommun. Signal Process. (TSP)*, Jul. 2019, pp. 406–410.

[12] A. C. Subrata, M. A. Riyadi, and T. Prakoso, "EEG-based BMI using multi-class motor imagery for bionic arm," in *Proc. 3rd Int. Conf. Mech., Electron., Comput., Ind. Technol. (MECnIT)*, Jun. 2020, pp. 255–260.

[13] M. N. Alam, M. I. Ibrahimy, and S. M. A. Motakabber, "Feature extraction of EEG signal by power spectral density for motor imagery based BCI," in *Proc. 8th Int. Conf. Comput. Commun. Eng. (ICCCE)*, Jun. 2021, pp. 234–237.

[14] J. Jin, T. Qu, R. Xu, X. Wang, and A. Cichocki, "Motor imagery EEG classification based on Riemannian sparse optimization and Dempster–Shafer fusion of multi-time-frequency patterns," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 58–67, 2023, doi: 10.1109/TNSRE.2022.3217573.

[15] Y. Miao et al., "Learning common time-frequency-spatial patterns for motor imagery classification," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 699–707, 2021.

[16] X. Geng et al., "A fusion algorithm for EEG signal processing based on motor imagery brain-computer interface," *Wireless Commun. Mobile Comput.*, vol. 2022, pp. 1–14, Mar. 2022.

[17] X. Gong, S. Chen, Y. Ban, and M. Wang, "Feature processing of multi-classification motor imagery EEG based on improved ICA and SVM," in *Proc. 2nd Int. Conf. Intell. Comput. Human-Comput. Interact. (ICHCI)*, Nov. 2021, pp. 318–321.

[18] X. Gao, D. Xu, M. Cheng, and S. Gao, "A BCI-based environmental controller for the motion-disabled," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 11, no. 2, pp. 137–140, Jun. 2003.

[19] Z. Wu and D. Yao, "Frequency detection with stability coefficient for steady-state visual evoked potential (SSVEP)-based BCIs," *J. Neural Eng.*, vol. 5, no. 1, pp. 36–43, Mar. 2008.

[20] C. Brunner, M. Naeem, R. Leeb, B. Graimann, and G. Pfurtscheller, "Spatial filtering and selection of optimized components in four class motor imagery EEG data using independent components analysis," *Pattern Recognit. Lett.*, vol. 28, no. 8, pp. 957–964, Jun. 2007.

[21] 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.*, vol. 32, no. 11, pp. 4814–4825, Nov. 2021.

[22] Y. Du, R. Xu, and J. Zhang, "Motor imagery analysis based on filter bank common spatial pattern," in *Proc. 2nd Int. Conf. Artif. Intell. Comput. Eng. (ICAICE)*, Nov. 2021, pp. 660–665.

[23] E. Dong, C. Li, L. Li, S. Du, A. N. Belkacem, and C. Chen, "Classification of multi-class motor imagery with a novel hierarchical SVM algorithm for brain-computer interfaces," *Med. Biol. Eng. Comput.*, vol. 55, no. 10, pp. 1809–1818, Oct. 2017.

[24] R. Chatterjee and T. Bandyopadhyay, "EEG based motor imagery classification using SVM and MLP," in *Proc. 2nd Int. Conf. Comput. Intell. Netw. (CINE)*, Jan. 2016, pp. 84–89.

[25] S. K. Mandal and M. N. B. Naskar, "Meta heuristic assisted automated channel selection model for motor imagery brain computer interface," *Multimedia Tools Appl.*, vol. 81, no. 12, pp. 17111–17130, May 2022.

[26] S. K. Mandal and M. N. B. Naskar, "Algorithmic analysis on automated channel selection framework for motor imagery BCI," in *Proc. 5th Int. Conf. Trends Electron. Informat. (ICOEI)*, Jun. 2021, pp. 32–39.

[27] A. Ramadhani, H. Fauzi, I. Wijayanto, A. Rizal, and M. I. Shapiai, "The implementation of EEG transfer learning method using integrated selection for motor imagery signal," in *Proc. 1st Int. Conf. Electron., Biomed. Eng., Health Inform.*, 2021, pp. 457–466.

[28] M. Rashid et al., "The classification of motor imagery response: An accuracy enhancement through the ensemble of random subspace k-NN," *PeerJ Comput. Sci.*, vol. 7, p. e374, Mar. 2021.

[29] B. Tasar and O. Yaman, "EEG signals based motor imagery and movement classification for BCI applications," in *Proc. Int. Conf. Decis. Aid Sci. Appl. (DASA)*, Mar. 2022, pp. 1425–1429.

[30] C. Uyanik, M. A. Khan, I. C. Brunner, J. P. Hansen, and S. Puthusserypady, "Machine learning for motor imagery wrist dorsiflexion prediction in brain-computer interface assisted stroke rehabilitation," in *Proc. 44th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2022, pp. 715–719.

[31] H. Altaheri et al., "Deep learning techniques for classification of electroencephalogram (EEG) motor imagery (MI) signals: A review," *Neural Comput. Appl.*, vol. 35, no. 20, pp. 14681–14722, 2021.

[32] W. Liu and Y. Zeng, "Motor imagery tasks EEG signals classification using ResNet with multi-time-frequency representation," in *Proc. 7th Int. Conf. Intell. Comput. Signal Process. (ICSP)*, Apr. 2022, pp. 2026–2029.

[33] Q. Wang, L. Wang, and S. Xu, "A novel motor imagery EEG classification approach based on time-frequency analysis and convolutional neural network," in *Recent Advances in AI-Enabled Automated Medical Diagnosis*. Boca Raton, FL, USA: CRC Press, 2022, pp. 329–346.

[34] J. F. Hwaidi and T. M. Chen, "Classification of motor imagery EEG signals based on deep autoencoder and convolutional neural network approach," *IEEE Access*, vol. 10, pp. 48071–48081, 2022.

[35] S. Chaudhary, S. Taran, V. Bajaj, and A. Sengur, "Convolutional neural network based approach towards motor imagery tasks EEG signals classification," *IEEE Sensors J.*, vol. 19, no. 12, pp. 4494–4500, Jun. 2019.

[36] X. Zhu, P. Li, C. Li, D. Yao, R. Zhang, and P. Xu, "Separated channel convolutional neural network to realize the training free motor imagery BCI systems," *Biomed. Signal Process. Control*, vol. 49, pp. 396–403, Mar. 2019.

[37] H. Wu et al., "A parallel multiscale filter bank convolutional neural networks for motor imagery EEG classification," *Frontiers Neurosci.*, vol. 13, p. 1275, Nov. 2019.

[38] S. Roy, K. McCreadie, and G. Prasad, "Can a single model deep learning approach enhance classification accuracy of an EEG-based brain-computer interface?" in *Proc. IEEE Int. Conf. Syst., Man Cybern. (SMC)*, Oct. 2019, pp. 1317–1321.

[39] Y. Li, X. Zhang, B. Zhang, M. Lei, W. Cui, and Y. Guo, "A channel-projection mixed-scale convolutional neural network for motor imagery EEG decoding," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 6, pp. 1170–1180, Jun. 2019.

[40] J. Z. Liu, F. F. Ye, and H. Xiong, "Convolutional neural network-based EEG signal recognition for multi-class motor imagery," *J. Zhejiang Univ.-Sci. A*, vol. 55, no. 11, pp. 2054–2066, 2021.

[41] Z. Y. Jia et al., "A method for motor imagery classification based on multi-scale feature extraction and squeeze-and-excitation model," *J. Comput. Res. Dev.*, vol. 57, no. 12, pp. 2481–2489, 2020.

[42] S. Woo, J. Park, J. Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.

[43] M. Tangermann et al., "Review of the BCI competition IV," *Frontiers Neurosci.*, vol. 6, p. 55, Jan. 2012.

[44] B. Blankertz et al., "The BCI competition III: Validating alternative approaches to actual BCI problems," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 14, no. 2, pp. 153–159, Jun. 2006.

[45] 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.

[46] 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.

[47] X. Deng, B. Zhang, N. Yu, K. Liu, and K. Sun, "Advanced TSGL-EEGNet for motor imagery EEG-based brain-computer interfaces," *IEEE Access*, vol. 9, pp. 25118–25130, 2021.

[48] R. Mane, N. Robinson, A. P. Vinod, S. Lee, and C. Guan, "A multi-view CNN with novel variance layer for motor imagery brain computer interface," in *Proc. 42nd Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2020, pp. 2950–2953.

[49] G. A. Altuwaijri, G. Muhammad, H. Altaheri, and M. Alsulaiman, "A multi-branch convolutional neural network with squeeze-and-excitation attention blocks for EEG-based motor imagery signals classification," *Diagnostics*, vol. 12, no. 4, p. 995, Apr. 2022.

[50] T. M. Ingolfsson et al., "EEG-TCNet: An accurate temporal convolutional network for embedded motor-imagery brain-machine interfaces," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2010, pp. 2958–2965.