EISATC-Fusion: Inception Self-Attention Temporal Convolutional Network Fusion for Motor Imagery EEG Decoding

Guangjin Liang, *Graduate Student Member, IEEE*, Dianguo Cao, Jinqiang Wang, Zhongcai Zhang, *Member, IEEE*, and Yuqiang Wu

***Abstract*— The motor imagery brain-computer interface (MI-BCI) based on electroencephalography (EEG) is a widely used human-machine interface paradigm. However, due to the non-stationarity and individual differences among subjects in EEG signals, the decoding accuracy is limited, affecting the application of the MI-BCI. In this paper, we propose the EISATC-Fusion model for MI EEG decoding, consisting of inception block, multi-head self-attention (MSA), temporal convolutional network (TCN), and layer fusion. Specifically, we design a DS Inception block to extract multi-scale frequency band information. And design a new cnnCosMSA module based on CNN and cos attention to solve the attention collapse and improve the interpretability of the model. The TCN module is improved by the depthwise separable convolution to reduces the parameters of the model. The layer fusion consists of feature fusion and decision fusion, fully utilizing the features output by the model and enhances the robustness of the model. We improve the two-stage training strategy for model training. Early stopping is used to prevent model overfitting, and the accuracy and loss of the validation set are used as indicators for early stopping. The proposed model achieves within-subject classification accuracies of 84.57% and 87.58% on BCI Competition IV Datasets 2a and 2b, respectively. And the model achieves cross-subject classification accuracies of 67.42% and 71.23% (by transfer learning) when training the model with two sessions and one session of Dataset 2a, respectively. The interpretability of the model is demonstrated through weight visualization method.**

***Index Terms*— Brain–computer interface (BCI), motor imagery (MI), attention collapse, temporal convolution network (TCN), transfer learning.**

I. Introduction

BRAIN-COMPUTER interface (BCI), a technology facilitating direct communication between the brain and external devices [1], finds extensive applications in various fields, including human-computer interaction, sports rehabilitation, and medical treatment for diseases [2], [3], [4]. Commonly used BCI paradigms are steady-state visual evoked potentials (SSVEP), P300, and motor imagery (MI) [5], with MI BCI being one of the most promising for application. MI BCI typically uses electroencephalogram (EEG) signals to detect motor imagery, allowing the user to control devices by imagining movements, such as moving electric wheelchairs, cursors, and upper-limb robots [6], [7], [8]. However, the instability of brain activity and the low signal-to-noise ratio (SNR) may yield diverse outcomes of EEG signals [9]. Moreover, the dependence on individual subjects and the correlation between EEG channels of MI EEG signals increase the complexity of analyzing and classifying brain signals [10].

At present, decoding of MI EEG signals primarily relies on traditional machine learning (ML) and deep learning (DL) [11]. Typically, traditional ML involves two steps process of feature extraction and classifier design. Deep learning provides an end-to-end approach that can automatically extract task-specific information from raw EEG signals without manually designed features. In the past five years, there has been a significant increase in the utilization of DL for classifying MI tasks. The convolutional neural network (CNN) is the most widely used architecture for MI classification [12]. Schirrmeister et al. [13] studied various deep CNN architectures, achieving comparable performance to conventional algorithms in EEG task decoding. Then, the compact EEGNet [14] extracts temporal and spatial features by designing the convolution kernel shape and achieves outstanding generalization to multiple paradigm datasets. TSFCNet [15] uses a simple network structure to extract excellent features and avoids the overfitting caused by complex network structures.

However, a single convolution pattern and convolution kernel size cannot effectively extract multi-scale advanced features. Therefore, the multi-scale network structure is used to solve this problem. MTFB-CNN [16] utilizes a parallel multi-scale time-frequency CNN block to adaptively extract EEG signal features in the temporal, frequency, and time-frequency domains. MSHCNN [17] uses 1D convolution to extract advanced temporal features and 2D convolution to extract temporal and spatial features. CMO-CNN [18] uses different filter scales and different branch depths to extract diverse and

Manuscript received 20 August 2023; revised 20 January 2024; accepted 20 March 2024. Date of publication 27 March 2024; date of current version 15 April 2024. This work was supported in part by Shandong Provincial Natural Science Foundation under Grant ZR2022MF236; and in part by the National Natural Science Foundation of China under Grant 62173207, Grant 62073187, and Grant 62073189. *(Corresponding author: Dianguo Cao.)*

The authors are with the College of Engineering, Qufu Normal University, Rizhao 276826, China (e-mail: lgj17852640119@163.com; caodg0318@163.com).

Digital Object Identifier 10.1109/TNSRE.2024.3382226

multi-level features for fusion. Zhao et al. [19] introduced a new 3D representation of EEG and a Multi-branch 3D CNN is designed to fully utilize the features on various dimensions of EEG. Similarly, inception [20] is a compact parallel structure that can efficiently extract multi-scale information. Incep-EEGNet [21] utilizes an inception-based architecture to decode the raw EEG signals. EEGSym [22] improves inter-subject MI classification performance with inception modules and residual connections.

The CNN cannot extract long temporal dependent features of time series data. Therefore, a CNN-based temporal convolutional network (TCN) [23] was proposed specifically for time series modelling and classification. EEG-TCNet [24] feeds the temporal features output by EEGNet into TCN to extract high-level long-term dependency information. ETC-Net [25] combines the efficient channel attention (ECA) and TCN components to extract channel features and temporal information. EEG-ITNet [26] uses inception block and TCN to extract rich spectral, spatial, and temporal information with less model complexity. The layer fusion of the model can capture the complex features of the input data and improve the representational capability of the model. TCNet-Fusion [27] adds layer fusion on EEG-TCNet to reduce feature loss and builds rich feature mappings. CCNN [28] fuses CNNs with different architectures and utilizes convolutional features at different layers to capture spatial and temporal features from raw EEG data.

Recently, the self-attention mechanism (SAM) [29] has been widely applied to EEG signal decoding. FB-Sinc-CSANet [30] introduces channel self-attention for local and global feature selection. SACNN-TFCSP [31] utilizes a self-attention-based CNN to extract the temporal and spatial information. CRAM [32] employs a recurrent SAM to investigate the temporal dynamics of the EEG signals while emphasizing the most discriminative temporal periods. The multi-head self-attention (MSA) [29] enables computing multiple global time-dependent features in parallel. ATCNet [33] uses MSA to highlight the most important information in EEG time series signals. Conformer [34] uses a MSA module to extract global long-term dependency features based on local temporal features extracted by CNN. TST-ICA [35] utilizes a temporal transformer and a spatial transformer to capture the temporal and spatial information, respectively. 3D DCSPNet [36] adaptively extracts optimal features from EEG signals through a spatial-spectral-temporal (SST) attention mechanism. However, when using MSA to decode EEG signals, it is easy to cause attention collapse due to the limited training set and the non-stationary characteristics of the EEG signals, as shown in Fig. 8. And it is difficult to illustrate the actual physical meaning of each attention head, which limits the interpretability of the model.

The main contributions of this paper are as follows:

1) We propose a high-performance, lightweight, and interpretable end-to-end MI EEG decoding model EISATC-Fusion. In the model, we design a DS Inception block by depthwise separable convolution to extract multi-scale frequency band information. And the cnnCosMSA module based on CNN and cos attention is designed to solve the attention collapse and improve the interpretability of the model. The TCN module is improved by the DS convolution, which greatly reduces the parameters of the model.

2) We improve the two-stage model training strategy, using the accuracy and loss of the validation set as indicators for the early stopping (ES). Experiments are conducted on multiple state-of-the-art models. The results show that the improved training strategy not only improves the decoding performance of the model but is also universal.

3) The transfer learning capability of EISATC-Fusion is studied, and the impact of learning rate and the amount of training data on model performance is discussed. The decoding performance of the cross-subject is further improved through transfer learning.

4) The interpretability of the model is illustrated through t-SNE and weight visualization methods, and the rationality of EISATCN-Fusion is demonstrated.

The rest of this paper is organized as follows: section II describes the proposed model and data preprocessing method, section III describes the experimental details and analysis of the results, and section IV summarizes our work. The model code can be obtained at https://github.com/LiangXiaohan506/EISATC-Fusion

II. Method

A. Overall Structure of EISATC-Fusion

EISATC-Fusion model consists of four modules: the EEGNet DS Inception (EDSI) module, cnnCos multi-head self-attention (cnnCosMSA) module, temporal depthwise separable convolutional network (TDSCN) module, and fusion module, as shown in Fig. 1.

The EDSI uses normal convolution and depthwise (DW) convolution to extract the temporal and spatial features and uses a depthwise separable (DS) inception block to extract the multi-scale time features. Then, the features with local information output by the EDSI are fed into the cnnCosMSA. The cnnCosMSA utilizes DW convolutional and cos attention to extract global features with long time-dependence. The features output by the EDSI and the cnnCosMSA are combined along the depth dimension and then fed into the TDSCN to extract high-level temporal features. The features output by the EDSI and the TDSCN are each fed into two fully connected (FC) layers, and the classification decision information output by the FC is fused through a learnable tensor. Finally, the model classification results are output through softmax.

B. Input Representation and Preprocessing

In this paper, the MI EEG signals $X_i \in \mathbb{R}^{C\times T}$ are fed into the proposed EISATC-Fusion model without applying filtering or removing artifacts. The signals include C channels and T sampling points. The output of model is $Y \in \mathbb{R}^{N_c}$, where N_c represents the number of classes.

The Z-score standardization is used to reduce the non-stationarity of the EEG signals as

$$x_0 = \frac{x_i - \mu}{\sqrt{\sigma^2}}, \tag{1}$$

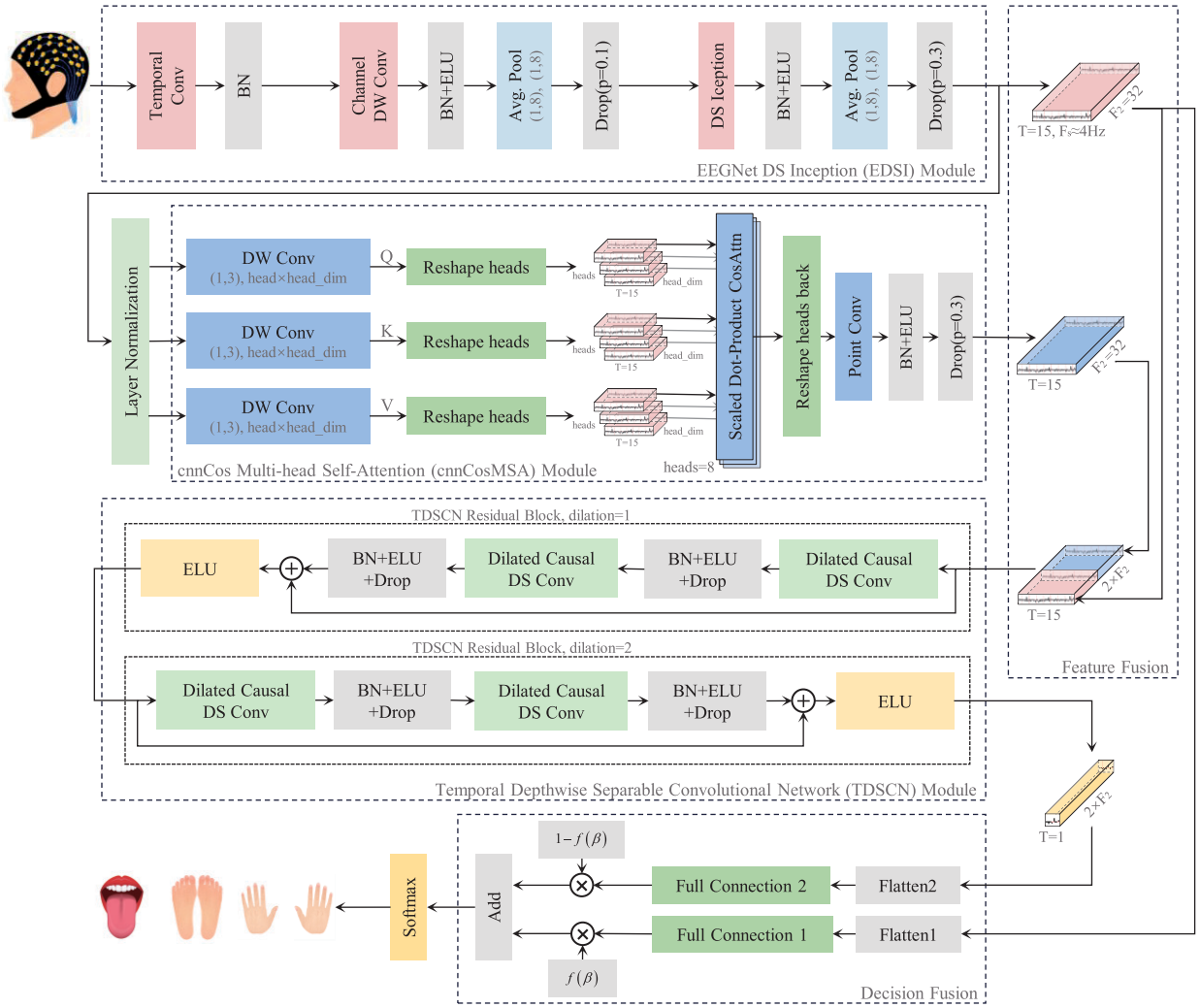

Fig. 1. The overall architecture of EISATC-Fusion, including an EEGNet DS Inception (EDSI) module, a cnnCos Multi-head Self-Attention (cnnCosMSA) module, a Temporal Depthwise Separable Convolutional Network (TDSCN) module, and a fusion module (feature fusion and decision fusion).

where x_0 and x_i represent the standardized output and training/test data, respectively. μ and σ^2 represent the mean and variance of the training data, and are utilized directly to normalize both the training and test data.

C. EEGNet DS Inception Module

The EEGNet DS Inception module is inspired by the EEGNet architecture introduced in [14], and the layer of SeparableConv2D in EEGNet is replaced by the DS Inception block. Moreover, the EDSI module employs different parameter values compared to those utilized in [14].

The core structure of the EDSI module is composed of three convolutional layers, and the detailed structure is shown in Fig. 1. The first layer is temporal convolution, using $F_1 = 16$ convolution kernels with a size of (1, 32) to learn temporal filters of different frequency bands. The second layer is channel convolution, using DW convolution with F_2 convolution kernels of size $(C, 1)$ and groups $= F_1$ to learn band-specific spatial filters. F_2 determines the size of the output features of the EDSI, and $F_2 = D \times F_1$, where D represents the number of connections between a filter in the previous layer and the filters of the current layer. Empirically, D is set to 2. The channel convolution is followed by an average pooling layer with a kernel size of (1, 8) and a step size of (1, 8), reducing the sampling rate of the signals to $\sim 32Hz$.

The last layer of convolution is the DS Inception (DSI) block, which is a multi-scale temporal convolution block. The DSI block contains three paths composed of DW convolution and a residual path, and at the end of the block there is a pointwise (PW) convolution layer, as shown in Fig. 2. The convolution kernel sizes of the first three paths are $(1, K_i)$, $(1, 2 \times K_i)$, and $(1, 4 \times K_i)$, where $K_i = 4$. The number of filters for each path is $F_2/4$, and the groups $= F_2/4$ for DW convolution. The fourth path adopts the structure of ResNet [37], and the maximum pooling layer with a kernel size of (3, 3) and a step size of (1, 1) is utilized for input information fusion. To maintain the dimensionality of the data passing through the DSI block, the maximum pooling layer is followed by the PW convolution layer with a kernel size of (1, 1), and the number of filters is $F_2/4$. The output features of the four paths are then connected in the depth dimension,

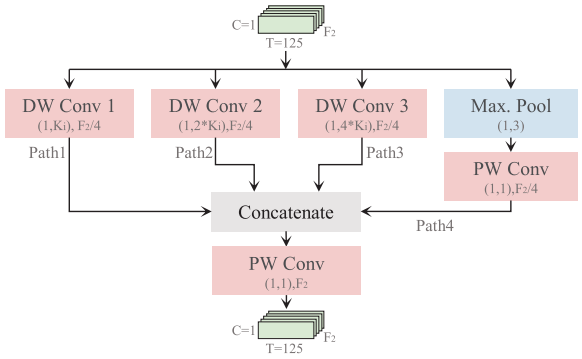

Fig. 2. Architecture of the DS Inception (DSI) block consisting of four paths.

and a PW convolution with F_2 filters performs information fusion in the depth dimension. The DSI is followed by an averaging pooling layer with a kernel size of (1, 8) and a step size of (1, 8), which downsamples the signal to $F_s \approx 4Hz$.

Following each convolutional layer, batch normalization (BN) [38] is applied to enhance the generalization of the model. The second and third BN layers are followed by an exponential linear unit (ELU) activation function for nonlinearity. Each pooling layer is followed by a dropout layer with a dropout rate of 0.5.

The output of the EDSI module is the time series of $Y_e \in \mathbb{R}^{d\times 1\times T_e}$, where $T_e = T/(8\times 8) \approx 15$ is the length of the output sequence, and $d = F_2$ is the dimension of the sequence.

D. cnnCos Multi-Head Self-Attention Module

The cnnCos Multi-head Self-Attention module is designed based on the CNN to address the issue of attention collapse during the processing of EEG signals by MSA. Cos attention is added to enhance the attention value and improve the interpretability of the model.

The attention mechanism is simulated by three components: query (Q), key (K), and value (V). The query/key/value vector are calculated by DW convolution, which has $heads \times head\_dim$ filters of size (1,3) and groups = d.

$$Q = DWConv_q(LN(Y_e)), \tag{2}$$

$$K = DWConv_k(LN(Y_e)), \tag{3}$$

$$V = DWConv_v(LN(Y_e)), \tag{4}$$

where LN is the layer normalization [39] and the $LN(Y_e)$ is the input sequence of the cnnCosMSA. Divide the Q, K, and V vectors into h subvectors, and take one subvector from each Q, K, and V vector to form an attention head. Then we obtain h attention heads, and according to experience, $h = heads = 8$, $head\_dim = 8$. The attention weights for each head are computed using scaled dot-product cosAttn, and the corresponding calculation process is illustrated in Fig. 3.

First, attention scores (ATscores) are calculated through Q and K:

$$ATscores(Q,K) = \frac{QK^T}{\sqrt{\sum Q^2}\sqrt{\sum K^2}}, \tag{5}$$

Then, the cos attention (CosAT) is calculated as

$$CosAT = 0.5cos(\omega' \cdot dis) + 0.5 \in [0,1], \tag{6}$$

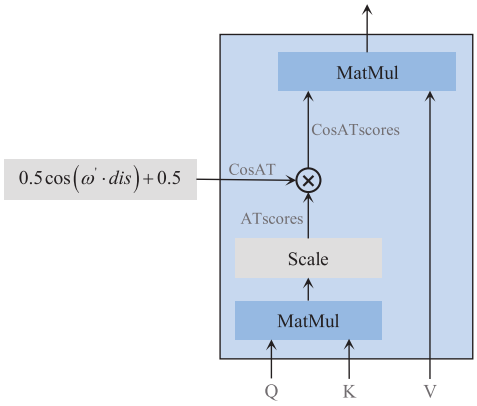

Fig. 3. Calculation process of scaled dot-product cosAttn.

where dis represents the distance between each moment in the input time series and is calculated as

$$dis_{m,n}^{i,j} = (i-m) + (j-n), \tag{7}$$

where $i = m = 1$, $j = n \in [1, T_e]$. The ω' represents the frequency as

$$\omega' = f(\omega)(T_e - 1) + 1 \in [1, T_e], \tag{8}$$

where $\omega \in \mathbb{R}^{1\times h}$ is a learnable tensor, $f(\cdot)$ represents the sigmoid function. The ATscore improved by CosAT is calculated as

$$CosATscore(Q,K) = CosAT \cdot ATscores(Q,K), \tag{9}$$

Finally, the output of the scaled dot-product cosAttn is obtained by multiplying CosATscore with V:

$$Attention(Q,K,V) = CosATscore \cdot V. \tag{10}$$

After the reshape operation, the dimension of $Attention$ is $[h \times head\_dim, 1, T\_e]$, which is different from the input dimension. Point convolution is used to keep the dimensionality of the input features constant, and is followed by BN and ELU. Dropout (p = 0.3) is used to prevent overfitting. The output of the cnnCosMSA module is $Y_m \in \mathbb{R}^{d\times 1\times T_e}$.

E. Temporal Depthwise Separable Convolutional Network Module

Temporal convolutional network does not need to explicitly maintain the state of sequence data, and thus have more efficient computation and longer time-dependent capabilities. In order to reduce the number of parameters of the TCN module without reducing the decoding performance of the module, we improved TCN by replacing dilated causal convolution with dilated causal DS convolution, which contains a layer of the dilated causal depthwise convolution and a layer of pointwise convolution. The detailed architecture of the TDSCN module is shown in Fig. 1.

The TDSCN module consists of two stacked residual blocks, each consisting of two dilated causal DS convolution layers followed by BN, ELU and dropout (p = 0.3) in sequence. Each residual block is followed by an ELU. A point convolution is used as a residual connection when the dilated causal DS

convolution changes the dimensions of the sequence fed into the residual block. In EISATC-Fusion, the dimensions of the sequence for input and output residual blocks are both $2 \times F_2$. Therefore, we use the identity mapping as the residual connection.

The dilation factor of the dilated causal DW convolution increases exponentially with the number of residual blocks L, i.e., the dilation factor of the i-th residual block is 2^{i-1}. Therefore, the receptive field size (RFS) of the TDSCN module as defined in

$$RFS = 1 + 2(K_t - 1)(2^L - 1), \tag{11}$$

where K_t is the kernel size of the convolution. In the EISATC-Fusion model, the input sequence length of the TDSCN is 15, and $L = 2$. Information is not ignored only if the RFS is greater than the length of the input sequence. Therefore, we set $K_t = 4$ (RFS=19>15) for all convolutional layers. The data fed into the TDSCN are described in the following subsection. The output of the TDSCN module is $Y_t \in \mathbb{R}^d$.

F. Fusion Module

The fusion module consists of two parts: feature fusion and decision fusion, as shown in Fig. 1.

Feature fusion combines the output of different layers in the model to extract the hidden information of the input data and improve the representation ability of the feature. The feature output by the EDSI are fine-grained descriptions of local information. The cnnCosMSA increases the global time-dependent property of the feature. Therefore, we fuse the features output by the EDSI and the cnnCosMSA module along the depth dimension to obtain a fusion feature $X_t \in \mathbb{R}^{(2\times d)\times T_t}$, where $T_t = T_m = T_e$. Then, the fusion feature is fed into the TDSCN to extract higher-level time-dependent information.

Decision fusion combines the outputs of multiple classifiers to reduce the uncertainty and errors of individual classifiers, improve the information integration ability of the model and allow the model to obtain more reliable decisions. The outputs of the EDSI and TDSCN are fed into the fully connected layer to obtain the prediction results $P_e \in \mathbb{R}^{N_C}$ and $P_t \in \mathbb{R}^{N_C}$, respectively. Then the learnable tensor β is used as the fusion coefficient, and the sigmoid function is used to limit the coefficient to $(0, 1)$. The fusion prediction result is calculated as

$$P = f(\beta) \cdot P_e + (1 - f(\beta)) \cdot P_t, \tag{12}$$

where $f(\cdot)$ is the sigmoid function.

To conclude, standardized MI EEG data are fed into the EISATC-Fusion model. The data are first processed through the EDSI and cnnCosMSA sequentially to extract features Y_e and Y_t respectively. The features are then fused and fed into the TDSCN. Then the output of the EDSI and cnnCosMSA are linearly transformed to obtain the prediction probabilities P_e and P_t, respectively. Finally, the two probabilities are adaptively fused and fed into the softmax function to obtain the classification result.

III. Experiments and Results

A. Datasets

Two famous MI EEG datasets, BCI Competitive IV 2a and BCI Competitive IV 2b, are used for model performance evaluation. Each dataset adopts different paradigms and has different numbers of samples.

1) BCI-2a: BCI Competition IV Dataset 2a [40] utilizes 22 electrodes to record the EEG of 9 subjects at a sampling rate of 250 Hz. Each subject is collected two sessions on two different days, with each session containing four different motor imagery tasks. The sample data for model training are collected within 0–4 seconds following the occurrence of a visual cue. The dimension of the data feed to the EISATC-Fusion is (C, T), where C = 22 and T = 1000. The $N_c = 4$.

2) BCI-2b: BCI Competition IV Dataset 2b [41] utilizes 3 electrodes to record the EEG of 9 subjects at a sampling rate of 250 Hz. Five sessions are collected for each subject, with left- and right-hand motor imagery tasks in each session. The sample data for model training are collected within 0–4 seconds following the occurrence of a visual cue. The dimension of the data feed to the EISATC-Fusion is (C, T), where $C = 3$ and $T = 1000$. The $N_c = 2$.

B. Experimental Details

The EISATC-Fusion model is built in the PyTorch 1.12 with Python 3.7 and trained by Nvidia GTX 3090 24 GB. Adam optimizer is used to optimize the model parameters with hyperparameter values: the learning rate $lr = 0.001$, β_1 is 0.9, β_2 is 0.999, and weight decay is 0.001. The cross-entropy function is used to calculate the overall model loss as

$$\ell(x, y) = \sum_{n=1}^{N} \frac{l_n}{N},$$
$$l_n = -log \frac{exp(x_{n,y_n})}{\sum_{c=1}^{N_C} exp(x_{n,c})}, \tag{13}$$

where x is the output of the model, y is the label, and N is the batch size.

The classification accuracy and kappa score (14) are utilized as evaluation metrics to assess the model performance.

$$kappa = \frac{p_0 - p_e}{1 - p_e}, \tag{14}$$

where p_0 represents the classification accuracy of the model and p_e represents the expected consistency level. The statistical significance is analyzed using the Wilcoxon signed-rank test.

Within-subject and cross-subject experiments are performed with the EISATC-Fusion model. For the within-subject experiment, the model is trained with the first session of BCI-2a and tested on the second session, and the model is trained with the first three sessions of BCI-2b and tested on the last two sessions. For cross-subject experiment, we use "leaving one subject out" (LOSO) evaluation method. One subject is selected from the dataset as the test set. Then the remaining subjects are used as the training set. Both within- and cross-subject experiments employ 5-fold cross-validation.

TABLE I
COMPARISON OF DECODING PERFORMANCE OF DIFFERENT METHODS ON BCI-2a

Method	S01	S02	S03	S04	S05	S06	S07	S08	S09	Average	Standard	k-score	p-value
EEGNet-8,2*[14]	84.72	57.64	89.58	61.11	70.49	57.99	81.25	80.90	74.65	73.15	12.01	0.6420	0.008
EEG-TCNet(Fixed) [24]	85.77	65.02	94.51	64.91	75.36	61.40	87.36	83.76	78.03	77.35	11.58	0.6978	0.011
TCNet-Fusion [27]	**90.74**	70.67	95.23	76.75	**82.24**	68.83	94.22	88.92	85.98	83.73	9.79	0.7778	0.953
SACNN-TFCSP [31]	85.76	62.50	87.15	76.04	78.82	59.72	92.36	86.46	84.72	79.28	11.35	-	0.028
Conformer [34]	88.19	61.46	93.40	78.13	52.08	65.28	92.36	88.19	88.89	78.66	15.30	0.7155	0.214
TSFCNet [15]	90.28	62.50	93.40	83.33	75.35	68.06	95.49	88.19	87.85	82.72	11.56	0.7695	0.373
MTFB-CNN [16]	90.52	68.10	93.97	74.14	80.17	72.41	**96.55**	**91.38**	**93.10**	84.48	10.80	0.7900	0.767
EISATC-Fusion	85.07	**73.26**	**95.49**	**87.15**	81.94	**73.96**	93.06	85.76	85.42	**84.57**	**7.48**	**0.7942**	

* Reproduced by ourselves.

TABLE II
COMPARISON OF DECODING PERFORMANCE OF DIFFERENT METHODS ON BCI-2b

Method	S01	S02	S03	S04	S05	S06	S07	S08	S09	Average	Standard	k-score	p-value
EEGNet-8,2*[14]	69.56	68.79	86.00	96.13	89.50	77.81	88.13	92.88	86.13	83.88	9.75	0.6776	0.008
DeepConvNet*[13]	71.63	68.21	83.13	95.38	92.75	84.75	90.88	91.13	87.00	84.98	9.41	0.6996	0.021
Conformer [34]	82.50	65.71	63.75	**98.44**	86.56	90.31	87.81	**94.38**	**92.19**	84.63	12.18	0.6926	0.515
TST-ICA [35]	81.69	**87.67**	74.64	96.60	90.74	85.91	88.73	84.06	83.09	85.90	6.17	-	0.441
MSHCNN [17]	**86.80**	77.94	65.97	97.97	93.24	**88.88**	86.80	82.89	86.80	85.25	9.19	-	0.594
TSFCNet [15]	76.25	70.00	83.75	97.50	92.81	86.56	88.44	92.50	89.69	86.39	8.63	0.7324	0.260
CMO-CNN [18]	86.38	77.27	79.44	97.29	94.25	88.33	89.79	86.31	85.69	87.19	**6.34**	-	0.767
EISATC-Fusion	75.00	72.86	**86.56**	96.88	**97.81**	84.38	**94.06**	93.75	86.88	**87.58**	9.07	**0.7515**	

* Reproduced by ourselves.

C. Model Training Strategy

We improve the two-stage training strategy proposed in [13], which only utilizes the accuracy on the validation set as the evaluation metric for ES. But when the model achieves the same accuracy, there may be different losses. A sample in BCI-2a is fed into the model, and the model will produce the probability that the sample falls into each class. Assuming that two different prediction probabilities, $p_1 = [0.24, 0.24, 0.24, 0.28]$ and $p_2 = [0.1, 0.1, 0.1, 0.7]$, are output by the model in different training epochs, the predicted result for both is class 3. Assuming that the true category of the sample is also 3, the losses calculated by (13) are 1.3564 and 0.9732 respectively. We prefer to obtain p_2, as it is more robust. Therefore, we improve the ES strategy and use loss and accuracy as the evaluation criteria. If the current training epoch obtains the best loss and the corresponding accuracy is higher than or equal to the previous best accuracy, then the ES epochs are reset.

The training process consists of two steps. In the first stage, the model is trained on the training set, and the best model is then obtained by applying the improved ES approach on the validation set. In the second stage, the model created in the first stage is trained again using both the training and validation sets. When the validation set loss is less than or equal to the minimum loss on the training set of the first stage, the second stage ends, and the model is saved.

D. Within-Subject Decoding Experiment

EISATC-Fusion is evaluated for within-subject decoding performance and compared with other state-of-the-art algorithms on BCI-2a and BCI-2b. The decoding accuracy (in percentage %), k-score and p-value are presented in Table I and Table II. The optimal data are highlighted.

The EISATC-Fusion is trained with the improved training strategy. The training epochs are 3000 for the first stage and 800 for the second stage. The ES epochs is 300. The batch size is 64.

The experimental results on BCI-2a are presented in Table I. EISATC-Fusion achieves the highest mean accuracy with the smallest standard deviation, and the accuracy on most subjects is higher than that of other algorithms. The average decoding accuracy of EISATC-Fusion exceeded that of CNN-based EEGNet-8,2 and TSFCNet by 15.61% ($p<0.01$) and 2.24%, respectively. These CNN-based models only focus on the local information of the EEG signal, while our model adds a SAM based on CNN to extract both local and global dependency information of the signal, which improves the decoding performance. Compared with SAM-based SACNN-TFCSP, it is improved by 6.67% ($p<0.05$). This proves that more attention heads can extract richer global information. And compared with MSA-based Conformer, it is improved by 7.51%. There is the attention collapse when applying MSA to decode EEG signal, and the cnnCosMSA solves the problem and further optimizes the original attention, improving the decoding performance of the model. Compared with TCN-based EEG-TCNet, it is improved by 9.33% ($p<0.05$). Compared with TCNet-Fusion based on model fusion and MTFB-CNN based on parallel multi-scale structure, they are only improved by 1% and 0.1% respectively, but the standard deviation is reduced by 23.60% and 30.74% respectively, which shows that our model has stronger individual adaptability.

TABLE III
ABLATION EXPERIMENT RESULTS OF DSI AND TDSCN ON BCI-2a

Method	Average	Standard	k-score	Parameters
EEGNet-16,2	76.43	12.43	0.6857	4,452
EDSI	**78.39**	12.36	0.7119	**5,476**
EEG-TCNet	81.33	9.17	0.7510	4,048
EEG+TDSCN	**81.67**	7.88	0.7557	**2,752**

TABLE IV
ABLATION EXPERIMENT RESULTS OF EISATC-FUSION ON BCI-2a

Removed module	Average	Standard	k-score	p-value
None(EISATC-Fusion)	**84.57**	7.48	0.7942	
CosAT	83.02	7.47	0.7737	0.038
cnnCosMSA	80.52	8.54	0.7402	0.012
TDSCN	77.89	9.63	0.7052	0.008
Feature Fusion	72.80	8.70	0.6373	0.008
Decision Fusion	82.60	7.54	0.7680	0.008
Fusion Module	68.75	10.01	0.5833	0.008

The experimental results on BCI-2b are shown in Table II, which shows similar results to those on BCI-2a. The decoding performance of EISATCN-Fusion is higher than models based on CNN, MSA and multi-scale structure. The improvement of the CMO-CNN based on the multi-branch structure, which has a higher performance than other models, is not obvious, and the average accuracy is only 0.45% higher. But the parameters of our model are only 26,374, which is 77.34% less than CMO-CNN (116,406).

E. Ablation Study

To investigate the effects of the DSI block and the TDSCN module, ablation experiments are performed on the EEGNet model and the EEG-TCNet model. The average decoding results for all subjects are shown in Table III. The training parameters are the same as subsection III-D.

The EEGNet-16,2 differs from the EDSI only in the third convolutional layer, all other parts are the same. And the accuracy of EEGNet-16,2 is higher than that of EEGNet-8,2 (75.58%, see the last column of the second row of Table V). The experimental results show that the accuracy of EDSI is improved by 2.56% compared to EEGNet-16,2, but the parameters of the model are increased by 23%. Since DSI adopts a parallel structure, the parameters of the model will inevitably increase. We mainly expected the DSI block to improve the accuracy of the model, so this is expected. EEG + TDSCN is obtained by replacing TCN with TDSCN in EEG-TCNet. The accuracy of EEG + TDSCN is 0.42% higher than that of EEG-TCNet, but the number of parameters of the model is reduced by 30.02%. It is well demonstrated that TDSCN greatly reduces the parameters of the model while guaranteeing the performance of the model.

To study the contribution of each module to the classification performance, we perform an ablation study on EISATC-Fusion. The results are shown in Table IV, and the best data are highlighted. The training parameters are the same as subsection III-D.

TABLE V
CLASSIFICATION PERFORMANCE OF DIFFERENT MODELS WITH DIFFERENT TRAINING STRATEGIES ON BCI-2a

Method	One stage		Two stages	
	Acc	Acc+Loss	Acc	Acc+Loss
EISATC-Fusion	76.04 ±10.89 (0.6806)	80.32 ±8.74 (0.7377)	77.04 ±8.95 (0.6939)	84.57 ±7.48 (0.7942)
EEGNet-8,2 [14]	73.15 ±12.01 (0.6420)	73.46 ±12.74 (0.6461)	75.04 ±12.60 (0.6672)	75.58 ±11.98 (0.6744)
ShallowConv [13]	74.04 ±11.30 (0.6538)	75.08 ±10.83 (0.6677)	76.70 ±11.24 (0.6893)	76.85 ±11.37 (0.6914)
DeepConv [13]	75.04 ±12.81 (0.6672)	75.58 ±12.17 (0.6744)	76.12 ±11.91 (0.6816)	76.23 ±13.80 (0.6831)
EEG-TCNet [24]	78.51 ±10.50 (0.7135)	81.33 ±9.17 (0.7510)	79.82 ±8.95 (0.7310)	82.02 ±9.65 (0.7603)

The results show that each module contributes to model decoding accuracy. The fusion module makes the greatest contribution to model performance. When the fusion module is removed the average decoding accuracy of the model decreases by 18.71% ($p<0.01$). The cnnCosMSA module and the TDSCN module also have a greater impact on the model. When they are removed from the model, the accuracy of the model decreases by 4.79% ($p<0.05$) and 7.90% ($p<0.05$) respectively. And CosAT also contributed to the performance of the model. When CosAT is removed from the model, the accuracy drops by 1.83% ($p<0.05$).

F. Comparing Different Training Strategies

The improved training strategy is tested on BCI-2a using the EISATC-Fusion and other commonly used algorithms. The results of the experiment are listed in Table V. The training parameters are the same as in subsection III-D.

For EISATC-Fusion, adding loss of the validation set as a indicator for ES improves the decoding accuracy of the model trained with one stage by 5.63% ($p=0.028<0.05$) and the model trained with two stages by 9.77% ($p=0.008<0.01$). This proves that using loss as the criterion for ES is effective, because it allows the features extracted by the model to have a larger inter-class distance and a smaller intra-class distance difference, improving the robustness of the model while maintaing accuracy. Whether using accuracy alone as an indicator of ES or using both loss and accuracy as indicators of ES, the accuracy of the model trained in two stages is higher than that in one stage. This proves that the two-stage training strategy can improve the decoding accuracy of the model. The second stage of training augments the data used for model training and allows the model to further extract effective features based on the first stage. And other algorithms also show the same result.

G. Cross-Subject Decoding Experiment

EISATC-Fusion is evaluated for cross-subject decoding performance and compared with other high-performance

TABLE VI
CROSS-SUBJECT CLASSIFICATION RESULTS OF DIFFERENT METHODS ON BCI-2a

Method	S01	S02	S03	S04	S05	S06	S07	S08	S09	Average	Standard	k-score	p-value
EEGNet-8,2*[14]	63.02	41.84	76.22	52.43	48.61	38.19	70.83	68.06	55.56	57.20	13.21	0.4293	0.011
ShallowConvNet*[13]	68.92	43.75	75.17	55.73	43.75	45.31	74.65	80.03	66.84	61.57	14.63	0.4876	0.008
CRAM [32]	61.02	42.35	73.11	50.43	50.74	51.48	67.26	69.72	66.85	59.22	10.13	-	0.011
CCNN [28]	62.07	42.44	63.12	52.09	49.96	37.16	62.54	59.32	69.43	55.35	10.66	-	0.008
Multi-branch 3D CNN [19]	49.51	40.74	64.50	44.56	54.29	40.46	58.87	59.75	56.83	52.17	8.76	0.4528	0.008
CMO-CNN [18]	68.75	44.44	**78.47**	55.90	53.12	51.56	67.70	76.38	73.78	63.34	12.31	-	0.038
3D DCSPNet [36]	67.23	50.31	73.82	**59.14**	58.08	**59.23**	71.75	**81.46**	59.79	64.53	9.72	-	0.173
EISATC-Fusion	**69.79**	**57.29**	76.04	58.85	**59.38**	51.22	**78.13**	81.08	**75.00**	**67.42**	10.85	0.5656	

* Reproduced by ourselves.

algorithms on BCI-2a. Table VI lists the accuracies for each subject and the average metrics for all subjects, with the best data highlighted.

Two sessions of each subject are used for training or testing. The EISATC-Fusion is trained with the improved training strategy. The training epochs are 3000 for the first stage and 800 for the second stage. The ES epochs is 100. The batch size is 128. The bias of full connection in decision module is set to false.

The results show that EISATC-Fusion has the best average decoding accuracy of 67.42%, and most subjects achieve the highest accuracy compared to other algorithms. The average decoding accuracy of EISATC-Fusion is improved by 17.87% ($p<0.05$) and 9.5% ($p<0.01$) compared to the CNN-based EEGNet-8,2 and ShallowConvNet, respectively. It is also improved by 13.85% ($p<0.05$) compared to CRAM based on MSA, and improved by 4.48% compared to 3D DCSPNet based on AM. Compared with CCNN based on model fusion, it is improved by 21.81% ($p<0.01$). And it is significantly improved ($p<0.05$) compared to Multi-branch 3D CNN and CMO-CNN, which are based on multi-branch. This proves that EISATC-Fusion has better generalization than other methods, and achieves higher performance when decoding new subjects in the practical application of MI BCI.

H. Transfer Learning Experiments for the Cross-Subject

Further, the transfer learning capability of the EISATC-Fusion model is studied. Use the first session of 8 subjects in BCI-2a to train the pre-training model. Randomly select different groups of data from the first session of the remaining subjects to fine-tune the model, each group of data contains one motor imagery trial of four classes, and the second session is used to test the performance of the model. During the model fine-tuning process, only the decision fusion module is trained, and the learning rate of transfer learning is $lr' = \alpha \times lr$, where α is the attenuation coefficient. The training parameters are the same as in subsection III-G.

The impact of the number of groups and learning rate of the fine-tuning model on the transfer learning is shown in Fig. 4. The 63.66% is the average classification accuracy of the pre-trained model. The classification accuracy of the model fine-tuned using one group of data is 33.83% (not shown in the figure) at $\alpha = 0.01$. When only one group of data is used for fine-tuning, the accuracy drops significantly compared to the pre-trained model, which shows that the model is prone to overfitting when the amount of data is insufficient. As the number of data groups increases, the average accuracy of the fine-tuned model increases, and when the number of groups reaches 9, the model achieves optimal decoding performance, which is 11.89% higher than the pre-trained model. The performance of the fine-tuned model is optimal when the original learning rate is used. As the learning rate decreases, the performance of the model also decreases. This demonstrates that a small learning rate easily makes EISATC-Fusion overfit.

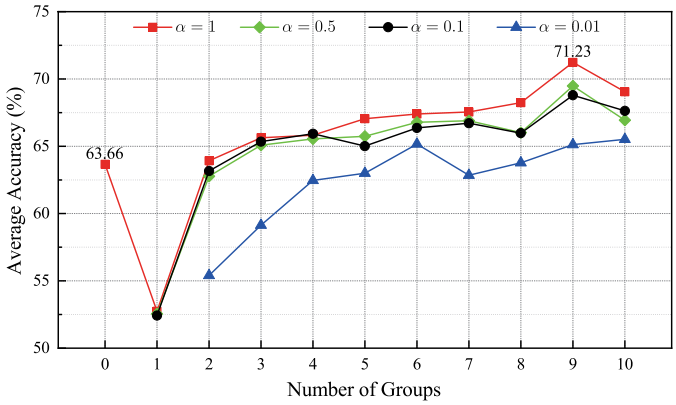

Fig. 4. The impact of the number of data groups and learning rate on transfer learning classification results.

We compared the transfer learning performance of three condition-based domain adaptation (DA) methods, including JDAO-Mix [42], DJDAN [43], and EA-CSP-LDA [44], and three fine-tuning-based methods, including DeepConvNet, MSFBCNN [45], and LSTM-MLP-T [46]. The experimental results are shown in Table VII. EISATC-Fusion improves the accuracy by 16.85% over the DA-based method, and it is also significantly improved compared to the fine-tuning-based method, except for LSTM-MLP-T. The accuracy of EISATC-Fusion is only 0.31% higher than LSTM-MLP-T, and the standard deviation of LSTM-MLP-T is smaller. This shows that LSTM-MLP-T is more robust than EISATC-Fusion in transfer learning, but EISATC-Fusion has more subjects with higher accuracy than LSTM-MLP-T.

I. Visualization

We illustrate the interpretability of the EISATC-Fusion model through two methods, including t-distributed stochastic

TABLE VII
CROSS-SUBJECT TRANSFER LEARNING RESULTS OF DIFFERENT ALGORITHMS ON BCI-2a

Method	Average	Standard	k-score
EISATC-Fusion	**71.23**	10.27	0.6253
JDAO-Mix (DA) [42]	60.69	-	-
DJDAN** (DA) [43]	53.20	-	-
EA-CSP-LDA* (DA) [44]	56.37	14.52	0.4702
DeepConvNet*(fine-tuning) [13]	57.37	12.62	0.4316
MSFBCNN (fine-tuning) [45]	60.00	-	-
LSTM-MLP-T (fine-tuning) [46]	71.01	7.49	-

* Reproduced by ourselves.
** Results from [42].

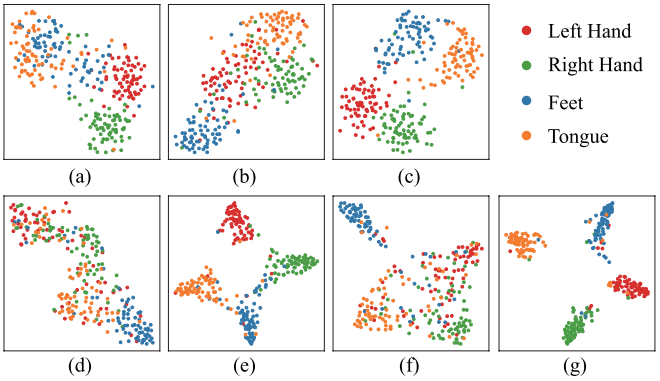

Fig. 5. t-SNE visualization illustrates the contribution of feature fusion for model performance. (a) output of the EDSI. (b) output of the cnnCosMSA. (c) fusion features. (d) output of the baseline model. (e) output of the baseline+FF. (f) output of the baseline+DF. (g) output of the baseline+FM.

neighbor embedding (t-SNE) feature visualization [47] and convolution kernel weight visualization.

The visualization of the feature distribution and the output of the model with and without feature fusion for subject 3 on BCI-2a are shown in Fig. 5. The EDSI + cnnCosMSA + TDSCN is defined as the baseline model. The first line in the figure is the output of the relevant modules in the EISATC-Fusion model. Fig. 5(a) is scattered, Fig. 5(b) is more concentrated, and the fusion features (Fig. 5(c)) with local and global information have better discriminative ability. Comparing Fig. 5(d) and Fig. 5(e), the distribution of model output is improved by feature fusion. Comparing Fig. 5(f) and Fig. 5(g), the output feature distribution of the model becomes scattered due to the removal of the feature fusion module. This all proves that the feature fusion module can improve the decoding performance of the model.

The visualization of the decision distribution for subject 7 on BCI-2a are shown in Fig. 6. The first line in the figure is the decision of the relevant modules in the EISATC-Fusion model. For subject 7, the fusion coefficient $\beta = -0.6480$, according to equation (12), the decision information of the EDSI module accounts for 34.34% of the fusion decision, which indicates that the decision information of the TDSCN module dominates the fused decision. Comparing Fig. 6(d) and Fig. 6(e), the decision distribution is improved by decision fusion. Comparing Fig. 6(f) and Fig. 6(c), the decision distribution becomes

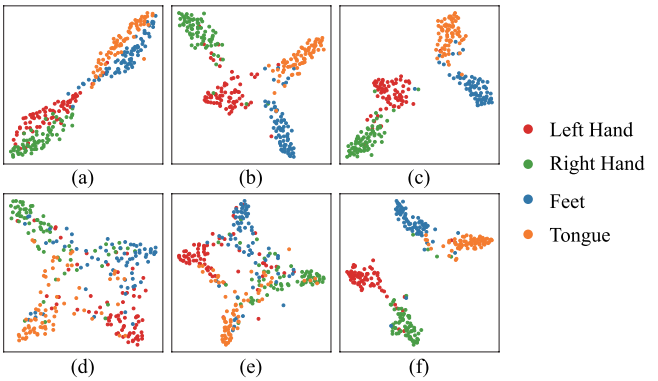

Fig. 6. t-SNE visualization illustrates the contribution of decision fusion for model performance. (a) decision of the EDSI. (b) decision of the TDSCN. (c) fusion decision. (d) decision of the baseline model. (e) decision of the baseline+DF. (f) decision of the baseline+FF.

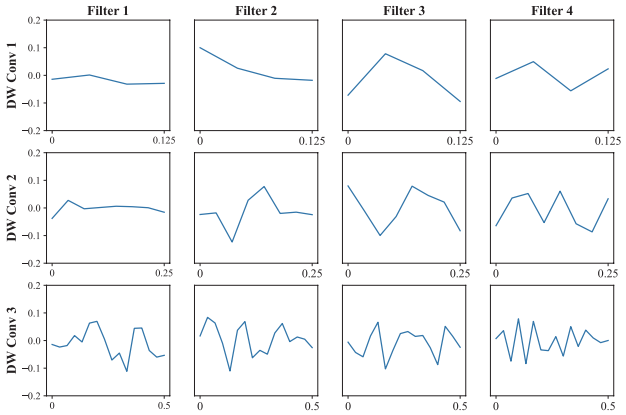

Fig. 7. Visualization of the convolutional weights of the DSI black for subject 1 in BCI-2a. Each row shows the temporal convolution kernel learned by a path.

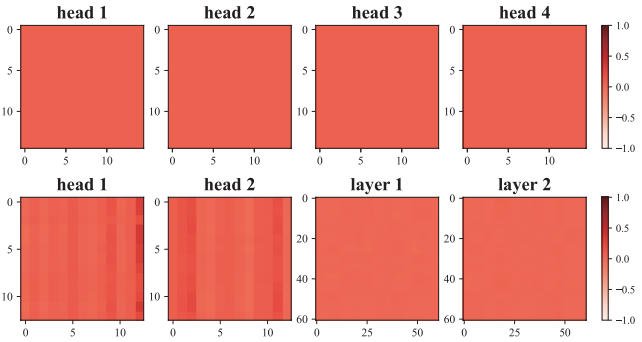

Fig. 8. Visualization of attention maps for subject 7 in BCI-2a. The first row is the attention maps of the 4 MSA heads. The left 2 of the second row are attention maps of the 2 heads of ATCNet. The right 2 of the second row are attention maps of the first 2 layer transformer of Conformer.

scattered due to the removal of the decision fusion module, but the change is not obvious. This is the same as the conclusion from Table IV, that decision fusion improves the performance of the models slightly less than feature fusion.

The convolution kernel weights of the DSI block are visualized, and the results are shown in Fig. 7. The first three paths of the DSI are shown, and 4 of the 32 convolution kernels are selected for each path. The convolution kernels in paths 1,

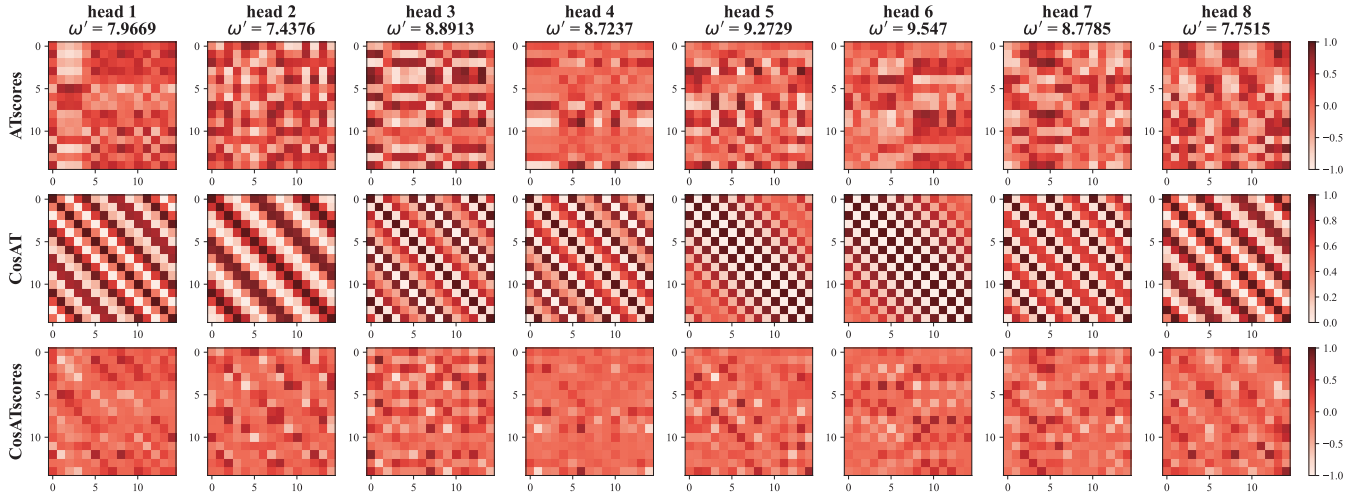

Fig. 9. Visualization of attention maps from 8 heads of cnnCosMSA module for subject 7 in BCI-2a.

2, and 3 have lengths of 4, 8, and 16 samples, corresponding to 0.125, 0.25, and 0.5 seconds in time, respectively, so their frequencies are estimated to be 8, 4, and 2 times the display period. Therefore, the frequency band of the time filter learned by path 1 is 0Hz-12Hz, path 2 is 0Hz-10Hz, and path 3 is 4Hz-10Hz. The frequency bands learned by the three paths are similar, but the path 1 is high and wide, and the path 3 is low and narrow. This demonstrates that the DSI block can learn multi-scale of band information with different sizes of temporal filters.

The attention maps of the three models are shown in Fig. 8. The top row shows the attention maps of the model that replaces cnnCosMSA in EISATC-Fusion with MSA, with 4 heads randomly selected from 8 heads. ATCNet [33] has only one layer of MSA with two heads, and all are displayed. One head was randomly selected from the 10 heads of the Transformer in the Conformer [34]. It can be clearly seen that the attention matrices tend to become uniform among patches (i.e., attention collapse). And as the complexity of the model increases or the number of MSA layers increases, the attention collapse becomes more obvious. This is because CNN converge more easily than MSA, especially with a small amount of training data, which causes the network to tend to learn from CNN when CNN is combined with MSA.

Three attention maps of the eight attention heads of the cnnCosMSA module are visualized, as shown in Fig. 9. The cnnCosMSA is a temporal attention module based on CNN that learns frequency information of EEG signals in the form of attention. It uses convolution to calculate attention and enables each attention head to learn filters of different frequencies through learnable tensors to achieve global filtering of EEG signals. The weight distribution of the original attention computed by (5) can be clearly observed from the first line of the figure, which is not collapsed. This indicates that the CNN-based cnnCosMSA module is fully learned. And the results showed that each attention head learned a different frequency. When the frequency is less than 8 (e.g., head 1, 2, and 8), the frequency characteristics of the attention of CosATscores improved by CosAT can be clearly observed. As the frequency increases, the global features are weakened and more local frequency features are presented, but the improvement of the original attention by CosAT is still obvious (enhanced useful attention weakened useless attention). The CosAT assigns a specific physical meaning to each attentional head, which improves the interpretability of the model.

IV. Conclusion

In this paper, we propose EISATC-Fusion, a high-performance end-to-end MI EEG decoding model. The model consists of four modules, and the ablation experiment demonstrates that each module of the model contributes to improving decoding performance. Furthermore, the two-stage training strategy is improved, and the comparative experiment demonstrates that the strategy enhances the decoding performance of the model and exhibits universality. We perform within-subject and cross-subject experiments on BCI-2a and BCI-2b using EISATC-Fusion and the training strategy. The transfer learning performance of the model is studied, and the decoding performance of the cross-subject is further improved through transfer learning. The interpretability of the model is illustrated through two visualization methods. However, we do not conduct online experiments and do not lighten the model. In future work, we further reduce the parameter count of the model and conduct online experiments.

References

[1] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain-computer interfaces for communication and control," *Clin. Neurophysiol.*, vol. 113, no. 6, pp. 767–791, 2002.

[2] S. Park, J. Ha, J. Park, K. Lee, and C.-H. Im, "Brain-controlled, AR-based home automation system using SSVEP-based brain-computer interface and EOG-based eye tracker: A feasibility study for the elderly end user," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 544–553, 2023.

[3] R. Zhang et al., "An adaptive brain-computer interface to enhance motor recovery after stroke," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 2268–2278, 2023.

[4] N. Grover, A. Chharia, R. Upadhyay, and L. Longo, "Schizo-Net: A novel schizophrenia diagnosis framework using late fusion multimodal deep learning on electroencephalogram-based brain connectivity indices," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 464–473, 2023.

[5] R. Abiri, S. Borhani, E. W. Sellers, Y. Jiang, and X. Zhao, "A comprehensive review of EEG-based brain-computer interface paradigms," *J. Neural Eng.*, vol. 16, no. 1, Jan. 2019, Art. no. 011001.

[6] M. Y. M. Naser and S. Bhattacharya, "Towards practical BCI-driven wheelchairs: A systematic review study," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 1030–1044, 2023.

[7] B. J. Edelman et al., "Noninvasive neuroimaging enhances continuous neural tracking for robotic device control," *Sci. Robot.*, vol. 4, no. 31, Jun. 2019, Art. no. eaaw684.

[8] Z. Tang, L. Zhang, X. Chen, J. Ying, X. Wang, and H. Wang, "Wearable supernumerary robotic limb system using a hybrid control approach based on motor imagery and object detection," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 1298–1309, 2022.

[9] W. Y. Hsu and Y. W. Cheng, "EEG-channel-temporal-spectral-attention correlation for motor imagery EEG classification," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 1659–1669, 2023.

[10] S. U. Amin, H. Altaheri, G. Muhammad, W. Abdul, and M. Alsulaiman, "Attention-inception and long-short-term memory-based electroencephalography classification for motor imagery tasks in rehabilitation," *IEEE Trans. Ind. Informat.*, vol. 18, no. 8, pp. 5412–5421, Aug. 2022.

[11] C. Gao, W. Liu, and X. Yang, "Convolutional neural network and Riemannian geometry hybrid approach for motor imagery classification," *Neurocomputing*, vol. 507, pp. 180–190, Oct. 2022.

[12] 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, Jul. 2023.

[13] R. T. Schirrmeister et al., "Deep learning with convolutional neural networks for EEG decoding and visualization," *Hum. Brain Mapping*, vol. 38, no. 11, pp. 5391–5420, Nov. 2017.

[14] 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, Jul. 2018, Art. no. 056013.

[15] H. Zhi, Z. Yu, T. Yu, Z. Gu, and J. Yang, "A multi-domain convolutional neural network for EEG-based motor imagery decoding," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 3988–3998, 2023.

[16] H. Li, H. Chen, Z. Jia, R. Zhang, and F. Yin, "A parallel multi-scale time-frequency block convolutional neural network based on channel attention module for motor imagery classification," *Biomed. Signal Process. Control*, vol. 79, Jan. 2023, Art. no. 104066.

[17] X. Tang, C. Yang, X. Sun, M. Zou, and H. Wang, "Motor imagery EEG decoding based on multi-scale hybrid networks and feature enhancement," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 1208–1218, 2023.

[18] X. Liu, S. Xiong, X. Wang, T. Liang, H. Wang, and X. Liu, "A compact multi-branch 1D convolutional neural network for EEG-based motor imagery classification," *Biomed. Signal Process. Control*, vol. 81, Mar. 2023, Art. no. 104456.

[19] 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.

[20] C. Szegedy et al., "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.

[21] M. Riyad, M. Khalil, and A. Adib, "Incep-EEGNet: A convnet for motor imagery decoding," in *Image and Signal Processing*, A. El Moataz, D. Mammass, A. Mansouri, and F. Nouboud, Eds. Cham, Switzerland: Springer, 2020, pp. 103–111.

[22] S. Pérez-Velasco, E. Santamaría-Vázquez, V. Martínez-Cagigal, D. Marcos-Martínez, and R. Hornero, "EEGSym: Overcoming inter-subject variability in motor imagery based BCIs with deep learning," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 1766–1775, 2022.

[23] S. Bai, J. Zico Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," 2018, *arXiv:1803.01271*.

[24] T. M. Ingolfsson, M. Hersche, X. Wang, N. Kobayashi, L. Cavigelli, and L. Benini, "EEG-TCNet: An accurate temporal convolutional network for embedded motor-imagery brain–machine interfaces," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2020, pp. 2958–2965.

[25] Y. Qin, B. Li, W. Wang, X. Shi, H. Wang, and X. Wang, "ETCNet: An EEG-based motor imagery classification model combining efficient channel attention and temporal convolutional network," *Brain Res.*, vol. 1823, Jan. 2024, Art. no. 148673.

[26] A. Salami, J. Andreu-Perez, and H. Gillmeister, "EEG-ITNet: An explainable inception temporal convolutional network for motor imagery classification," *IEEE Access*, vol. 10, pp. 36672–36685, 2022.

[27] Y. K. Musallam et al., "Electroencephalography-based motor imagery classification using temporal convolutional network fusion," *Biomed. Signal Process. Control*, vol. 69, Aug. 2021, Art. no. 102826.

[28] S. U. Amin, M. Alsulaiman, G. Muhammad, M. A. Mekhtiche, and M. S. Hossain, "Deep learning for EEG motor imagery classification based on multi-layer CNNs feature fusion," *Future Gener. Comput. Syst.*, vol. 101, pp. 542–554, Dec. 2019.

[29] A. Vaswani et al., "Attention is all you need," in *Proc. NIPS*, 2017, pp. 6000–6010.

[30] J. Chen, D. Wang, W. Yi, M. Xu, and X. Tan, "Filter bank sinc-convolutional network with channel self-attention for high performance motor imagery decoding," *J. Neural Eng.*, vol. 20, no. 2, Mar. 2023, Art. no. 026001.

[31] R. Zhang, G. Liu, Y. Wen, and W. Zhou, "Self-attention-based convolutional neural network and time-frequency common spatial pattern for enhanced motor imagery classification," *J. Neurosci. Methods*, vol. 398, Oct. 2023, Art. no. 109953.

[32] D. Zhang, L. Yao, K. Chen, and J. Monaghan, "A convolutional recurrent attention model for subject-independent EEG signal analysis," *IEEE Signal Process. Lett.*, vol. 26, no. 5, pp. 715–719, May 2019.

[33] H. Altaheri, G. Muhammad, and M. Alsulaiman, "Physics-informed attention temporal convolutional network for EEG-based motor imagery classification," *IEEE Trans. Ind. Informat.*, vol. 19, no. 2, pp. 2249–2258, Feb. 2023.

[34] Y. Song, Q. Zheng, B. Liu, and X. Gao, "EEG conformer: Convolutional transformer for EEG decoding and visualization," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 710–719, 2022.

[35] A. Hameed et al., "Temporal–spatial transformer based motor imagery classification for BCI using independent component analysis," *Biomed. Signal Process. Control*, vol. 87, Jan. 2024, Art. no. 105359.

[36] Y. Wen, W. He, and Y. Zhang, "A new attention-based 3D densely connected cross-stage-partial network for motor imagery classification in BCI," *J. Neural Eng.*, vol. 19, no. 5, Oct. 2022, Art. no. 056026.

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, vol. 16, 2016, pp. 770–778.

[38] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[39] J. Lei Ba, J. Ryan Kiros, and G. E. Hinton, "Layer normalization," 2016, *arXiv:1607.06450*.

[40] 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, Tech. Rep., Jan. 2008. [Online]. Available: https://www.bbci.de/competition/iv/desc_2a.pdf

[41] R. Leeb, C. Brunner, G. Müller-Putz, A. Schlögl, and G. Pfurtscheller, "BCI competition 2008-Graz data set B," Inst. Knowl. Discovery, Lab. Brain-Comput. Interfaces, Graz Univ. Technol., Graz, Austria, Tech. Rep., 2008, pp. 1–6. [Online]. Available: https://www.bbci.de/competition/iv/desc_2b.pdf

[42] P. Chen, H. Wang, X. Sun, H. Li, C. Grebogi, and Z. Gao, "Transfer learning with optimal transportation and frequency mixup for EEG-based motor imagery recognition," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 2866–2875, 2022.

[43] X. Hong et al., "Dynamic joint domain adaptation network for motor imagery classification," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 556–565, 2021.

[44] H. He and D. Wu, "Transfer learning for brain–computer interfaces: A Euclidean space data alignment approach," *IEEE Trans. Biomed. Eng.*, vol. 67, no. 2, pp. 399–410, Feb. 2020.

[45] 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.

[46] M. Dehghani, A. Mobaien, and R. Boostani, "A deep neural network-based transfer learning to enhance the performance and learning speed of BCI systems," *Brain-Comput. Interfaces*, vol. 8, nos. 1–2, pp. 14–25, Apr. 2021.

[47] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 86, pp. 2579–2605, 2008.