An Auxiliary Synthesis Framework for Enhancing EEG-Based Classification With Limited Data

Sui Liang, Shaolong Kuang, Deheng Wang, Zhaohua Yuan,
Hongmiao Zhang, *Member, IEEE*, and Lining Sun

***Abstract*— While deep learning algorithms significantly improves the decoding performance of brain-computer interface (BCI) based on electroencephalogram (EEG) signals, the performance relies on a large number of high-resolution data for training. However, collecting sufficient usable EEG data is difficult due to the heavy burden on the subjects and the high experimental cost. To overcome this data insufficiency, a novel auxiliary synthesis framework is first introduced in this paper, which composes of a pre-trained auxiliary decoding model and a generative model. The framework learns the latent feature distributions of real data and uses Gaussian noise to synthesize artificial data. The experimental evaluation reveals that the proposed method effectively preserves the time-frequency-spatial features of the real data and enhances the classification performance of the model using limited training data and is easy to implement, which outperforms the common data augmentation methods. The average accuracy of the decoding model designed in this work is improved by (4.72±0.98)% on the BCI competition IV 2a dataset. Furthermore, the framework is applicable to other deep learning-based decoders. The finding provides a novel way to generate artificial signals for enhancing classification performance when there are insufficient data, thus reducing data acquisition consuming in the BCI field.**

***Index Terms*— Brain computer interface, electroencephalogram, motor imagery, deep learning, data augmentation.**

I. Introduction

BRAIN-COMPUTER interface (BCI) identifies brain activity and converts it into instructions or information, and establishes pathways between the brain and external devices [1]. Electroencephalogram (EEG) is one of the commonly used brain activity recording methods for BCI. EEG measures the scalp electrical signals generated by the brain, and has the characteristics of high temporal resolution, low trauma and low cost [2], [3]. BCI system based on EEG is usually used to realize prosthesis control [4], emotion recognition [5], speech recognition [6], epilepsy prediction [7], sleep monitoring [8], etc. However, limited by the data acquisition and low recognition accuracy, the practical application of BCI technology is still challenging.

Over the past decades, numerous studies have applied deep learning methods to EEG signal recognition [9], thus the classification performance has been greatly improved. Deep learning methods automatically extract features from original signals and complete classification [10]. But this kind of methods usually need a large number of training data to learn the latent features, small datasets and low-resolution data easily tend to cause overfitting and feature dependence of models [11]. Eventually, the classification performance of deep learning methods may even be inferior to that of traditional methods, such as linear discriminant analysis, support vector machine, naive Bayes classifier, etc. Some few-shot learning strategies, such as multi-task learning, transfer learning and meta-learning [12], try to solve this problem from the aspects of models and algorithms, and have achieved some results, but it is complicated to design such algorithms. By contrast, studying from the perspective of data, it is expected to fundamentally solve the training problem of deep learning network caused by insufficient data. Compared with the computer vision (CV) and natural language processing (NLP), it is difficult to collect enough high-quality data in the BCI field [13], [14]. There are generally four reasons: 1) Data collection experiment is cumbersome and takes a long time. Subjects may feel uncomfortable during the collection process, and the state of subjects will affect the quality of the data [15]. 2) Because of the physical dysfunction of some subjects, information (such as movement and sound) is difficult to track [16]. 3) The collected data will also be discarded due to problems such as interference and missing information [15], [17]. 4) Scarcity of qualified subjects and experimental environment due to the strict requirements [18].

Data augmentation is one of the effective ways to alleviate the problem of insufficient data [19]. This approach is based on the assumption that more information can be extracted from the original dataset through augmentation, and it artificially increases the size of training dataset by

Manuscript received 13 October 2022; revised 29 December 2022 and 22 March 2023; accepted 14 April 2023. Date of publication 20 April 2023; date of current version 28 April 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62127810. *(Corresponding author: Hongmiao Zhang.)*

Sui Liang, Zhaohua Yuan, Hongmiao Zhang, and Lining Sun are with the Jiangsu Provincial Key Laboratory of Advanced Robotics, School of Mechanical and Electric Engineering, Soochow University, Suzhou 215000, China (e-mail: zhanghongmiao@suda.edu.cn).

Shaolong Kuang is with the College of Health Science and Environmental Engineering, Shenzhen Technology University, Shenzhen 518118, China (e-mail: kuangshaolong@gmail.com).

Deheng Wang is with the Department of Physiology, School of Basic Medical Sciences, Shanghai University of Traditional Chinese Medicine, Shanghai 201203, China (e-mail: wangdeheng@shutcm.edu.cn).

Digital Object Identifier 10.1109/TNSRE.2023.3268979

deforming or sampling. [20]. Data augmentation technology has been mature in the field of CV, however, the geometric transformations that used for image data augmentation may not be applicable to BCI research due to the characteristics of EEG signals, such as non-stationarity, time-varying sensitivity, individual differences, etc. [21], [22], [23]. According to the investigations of Lashgari and He et al. [20], [24], common augmentation methods used in this area include cropping (sliding window), adding noise, and generative adversarial networks (GAN) [25]. Other methods such as recombination of segmentation [26], Fourier transform [27], synthetic minority over-sampling technique (SMOTE) can also be found in some BCI studies [28].

Cropping is a simple and effective method used for EEG data augmentation. This approach uses a sliding window to slice raw data for getting many more training samples. Schirrmeister et al. [29] used 2s sliding window to crop and expand the original motor imagery data, and improved the decoding performance of deep convolutional network. Zhao et al. [30] trained a network composed of three branches of deep convolutional neural network by cropping method, and alleviated the overfitting phenomenon. Mousavi et al. [31] proposed an automatic sleep stages recognition method based on deep learning, and used the cropping method to balance the data of different sleep stages, finally obtained 93.55% accuracy, which higher than using GAN augmentation method. In addition, Luo [32], Tayeb [33], Majidov [34] et al. also improved the classification performance by cropping method.

Another easily implemented augmentation method is adding noise to the raw data. It achieves expansion by adding noise to the original EEG signals or extracted feature maps. Wang et al. [35] effectively improved emotion recognition performance of deep learning model by adding Gaussian noise to DE features. Salama et al. [36] added the Gaussian noise signals with zero mean and unit variance to the raw data to improve their 3D-CNN emotion recognition performance, and the accuracy increased from 79.11% and 79.22% to 88.49% and 87.44% for valence and arousal classification, respectively. Li et al. [37] proposed a CP-MixedNet and used the amplitude-perturbation data augmentation method to train the model, this method added noise to the amplitudes of spectral images, and the classification performance on the BCI Competition IV 2a dataset and the High gamma dataset was significantly improved.

Despite the advances of cropping and adding noise in the field of EEG data augmentation, these two methods still cannot completely satisfy the needs of artificial multi-channel EEG signals generation, due to information loss, redundant noise, or inability to use underlying features of data [38]. Data augmentation methods based on deep learning can realize feature extraction to reconstruct artificial data [39]. GAN is one of the most common methods, which aims at achieving Nash equilibrium [40] between generative model and discriminative model, learning distributions from original data and generating new data [41]. Luo et al. [15] used conditional Wasserstein GAN (cWGAN) and selective Wasserstein GAN (sWGAN) to improve the performance of the classifiers in their emotion recognition task. Nik Aznan et al. [42] used Deep Convolutional GAN (DCGAN), Wasserstein GAN (WGAN) and Variational Autoencoder (VAE) models respectively to generate artificial data and improve the decoding performance of the models across subjects. Xu et al. [23] designed a BWGAN-GP model to improve the class imbalance, and the area under the curve tested with EEGNet was 3.7% higher than the original data. Xu et al. [43] apply a GAN based on convolutional neural network (CNN) and recurrent neural network (RNN) to synthesize artificial multichannel EEG preictal samples for ES prediction, and the accuracy and area under the curve improve from 73.0% and 0.676 to 78.0% and 0.704.

Although GAN-based methods perform well in EEG data augmentation, there are still some problems, such as many variant models, complex training process and instability [17], [44], and an additional decoding model is usually required to complete the final classification task. In this study, we propose a data synthesis framework based on deep generative model to improve classification performance. The framework uses limited real data and Gaussian noise to synthesize artificial data, and is expected to reduce the complexity of training progress while preserving the feature information of real data and reducing the redundant noise of synthesized data. The novelties of this study are summarized as follows:

- An auxiliary approach is introduced to design a synthesis framework, which utilize a pre-trained decoding model to assist in synthesizing artificial signals.
- A decoding model and a generative model are designed to extract the temporal-spatial features of EEG signals for classification and synthesis.
- Different number of training samples is set to explore the improvement of the decoding performance under limited data. The method is transferred and implemented to the state-of-the-art decoders.
- Visualization methods of multiple perspectives are provided for interpreting the artificial data and framework.

The rest of this paper is organized as follows. Section II introduces the dataset, describes the proposed framework and general methods used for comparison and evaluation. Section III presents the experimental results. Section IV performs discussions. Section V gives the main conclusions.

II. Materials and Methods

A. Dataset and Preprocessing

The BCI Competition IV 2a is used in this study [45]. This dataset contains EEG signals from nine subjects when they imagine the movements of left hand, right hand, foot and tongue. Each subject requires to complete 2 sessions, each of which contains 288 trials motor imagination tasks. EEG Signals are collected through 25 Ag/AgCl electrodes, the first 22 are EEG channels, and the last 3 are EOG channels. The sampling frequency is 250 Hz, and a band-pass filter of 0.5-100 Hz and a notch of 50 Hz are implemented. In this study, only the EEG channels are focused, and the data of cue and motor imagery periods are clipped as a single sample, each sample lasts for 4s. Finally, the shape of the dataset for

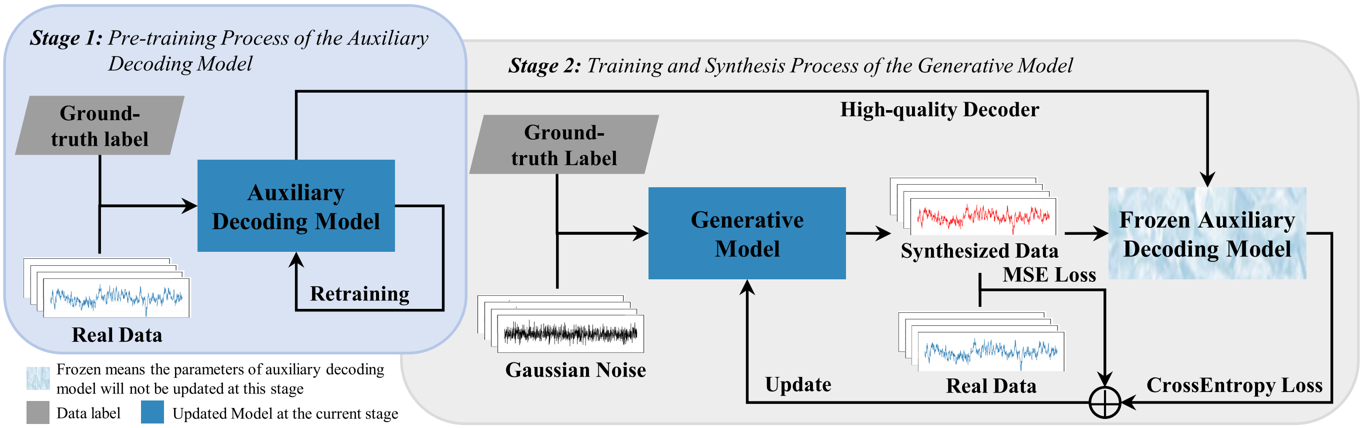
Stage 1: Pre-training Process of the Auxiliary Decoding Model
Ground-truth label
Auxiliary Decoding Model
Retraining
Real Data
Stage 2: Training and Synthesis Process of the Generative Model
High-quality Decoder
Ground-truth Label
Generative Model
Synthesized Data
Frozen Auxiliary Decoding Model
MSE Loss
Gaussian Noise
Update
Real Data
CrossEntropy Loss
Frozen means the parameters of auxiliary decoding model will not be updated at this stage
Data label
Updated Model at the current stage

Fig. 1. Flow chart of the auxiliary synthesis framework.

each subject is 576 × 22 × 1000. All datasets are normalized before inputting to the model.

B. Auxiliary Synthesis Framework

In order to effectively retain the original information and reduce redundant noise, the proposed framework is built based on deep learning methods which have been proven to work for EEG synthesis and classification. In the following subsections, the generic layout of the auxiliary synthesis framework will be described first, follow by the details of auxiliary decoding model, generative model, loss function and training configuration.

1) Framework Overview: An architecture overview of the auxiliary synthesis framework is presented in Fig. 1. The architecture of this framework can be divided into 2 stages:

- Pre-training Process of the Auxiliary Decoding Model. In this stage, the auxiliary decoding model is pre-trained by the limited real samples, ensure that the accuracy of the decoding model is not less than the given threshold, which can be determined by cross-validation on the real samples. This option prevents the performance degradation of the decoding model caused by random factors during the training process, thereby improving the stability of the generative model and further improving the quality of the synthesized data. In fact, training with insufficient data is likely to produce an unstable result [46].
- Training and Synthesis Process of the Generative Model. The pre-trained auxiliary decoding model is used to assist the generative model in training and synthesizing new data. Specifically, in this stage, the generative model learns the mappings between labeled Gaussian noise and real data distribution to synthesize specific artificial data, and then the synthesized data and their ground-truth labels will be input into the auxiliary decoding model to obtain the probability distribution, which is used to calculate the cross-entropy (CE) loss. The mean squared error (MSE) loss between synthesized data and real data is also calculated. Finally, both CE and MSE are used to optimize the generative model. The parameters of the auxiliary decoding model are frozen at this stage, only the generative model is updated. All synthesized samples from the last epoch of the training stage are retained and labeled with the ground-truth labels, and eventually appended into the training dataset to retrain the decoding model.

2) Auxiliary Decoding Model: The auxiliary decoding model captures the latent features of EEG data and output the probability distribution. In the framework, it is used to help the generative model synthesize artificial data. It is also used for final classification tasks. As shown in Fig. 2(a), a combination of ordinary convolution and depthwise separable convolution is used to extracts the spatial-temporal features of real data or synthesized data. Depthwise separable convolution reduces the parameters while maintain the decoding performance simultaneously [47], [48]. Considering the efficacy and generalizability of deep learning on EEG-based decoding of motor imagery, the Squeeze-and-Excitement (SE) attention mechanism is added to improve the classification performance by changing the weights of different channels [27], [49]. These weighted spatial-temporal features are finally classified by a fully connected layer.

3) Generative Model: Generative model learns real data distributions, and synthesizes new data from a batch of fixed Gaussian noise. The architecture is shown in Fig. 2(b). This model is built based on transposed convolution, which enable the neural network to learn how to up-sample in the best way, and improves the quality of synthesized data [50], [51], [52].

Gaussian noise and category labels are used as inputs for the generative model. The label is first encoded by embedding layer, then the encoded label is regarded as a new channel to concatenate with the Gaussian noise, and finally the concatenated data are transformed by transposed convolution layers from time and spatial directions. In order to maintain the size of synthesized data consistent with the original data, we set different stride sizes in the transposed convolution operations and add padding operations, the specific parameters are shown in Fig. 2(b).

4) Loss Functions: Inspired by the loss function used in [53], [52], and [55], both cross-entropy loss and mean

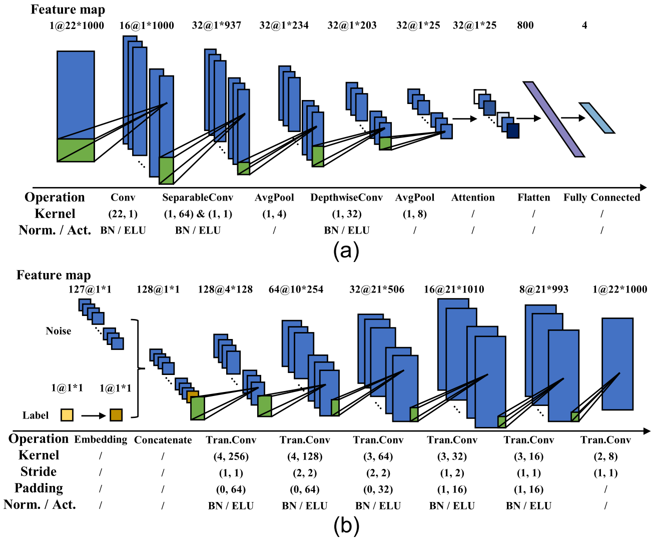

Fig. 2. Architecture of the neural network used in auxiliary synthesis framework. (a) Auxiliary decoding Model, (b) Generative model. Conv: Convolution, Tran.Conv: Transposed convolution, Norm.: Normalization, Act.: Activate function, BN: Batch normalization, ELU: Exponential linear unit.

squared error loss are used to optimize the auxiliary synthesis framework. The cross-entropy loss enables the generative model to focus on the classification features of the data, and ensures that the synthesized data maintain a certain level of classification performance under the current decoding model [55]. At the generative model training stage, the cross-entropy loss of the synthesized data is calculated using the probability distribution that is provided by the pre-trained auxiliary decoding model, and is defined as follow:

$$L_{CE} = \frac{1}{N}\sum_{n=1}^{N} -\log \frac{\exp\left(A\left(G\left(z_n, y_{r,n}\right)\right)\left[y_{r,n}\right]\right)}{\sum_{c=0}^{C-1}\exp\left(A\left(G\left(z_n, y_{r,n}\right)\right)[c]\right)} \tag{1}$$

where N is the batch size, C is the number of classes, z_n is the gaussian noise, $y_{r,n}$ is the label of real data, $G(\bullet)$ is the generative model, $A(\bullet)$ is the auxiliary decoding model.

The mean squared error loss is used in the final loss to ensure that the distribution of the synthesized data is similar to that of the real data, and to prevent the generative model from synthesizing unexpected data, and it is defined as follow:

$$L_{MSE} = \frac{1}{N}\sum_{n=1}^{N}\left(x_{r,n} - G\left(z_n, y_{r,n}\right)\right) \tag{2}$$

where N is the batch size, $x_{r,n}$ is the real data, z_n is the gaussian noise, $y_{r,n}$ is the label of real data, $G(\bullet)$ is the generative model.

The final loss used to update the generative model consists of the cross-entropy loss and the mean squared error loss, as shown in Fig. 1, and is defined as follow:

$$L_{total} = \alpha L_{MSE} + \beta L_{CE} \tag{3}$$

where α and β are weights that control the interaction of the losses, we set α to 1 and β to 0.0001 in this study.

TABLE I
ARCHITECTURE OF THE DISCRIMINATOR USED IN CGAN

Block	Options	Kernel	Filters	Norm./Act.
Input	Embedding Concatenate			
Block1	Conv2D	(1, 128)	8	BN/ELU
Block2	Conv2D Dropout	(22, 1)	16	BN/ELU
Block3	Conv2D Dropout	(1, 64)	32	BN/ELU
Block4	Conv2D Dropout	(1, 32)	64	BN/ELU
Output	FC FC		64	ELU Sigmoid

Norm.: Normalization, Act.: Activate function, BN: Batch normalization, ELU: Exponential linear unit, FC: Fully connected

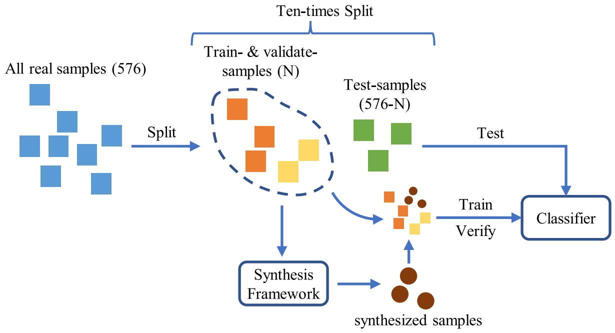

Fig. 3. Diagram of cross-validation analysis. N: Number of limited real samples (i.e., 40, 80, . . . , 520).

5) Detailed Configuration of Training: All models are designed based on Pytorch and are trained and tested using an NVIDIA RTX A5000. Adam optimizer is used for both decoding model and generative model training. We set the weight decay coefficient to 0.001, and use early stopping strategy [56] when training the auxiliary decoding model. Early stopping strategy reduces the excessive influence of incoherent gradients and improves the generalization ability of the model. Both models are trained using a learning rate decay strategy, and the initial learning rate is set to 0.0003.

C. General Methods Used for Comparison

Following methods are chosen for performance comparison.

1) Cropping: Cropping is commonly used for EEG data augmentation. In our experiment, sliding window with 3.9s length and step with 0.1s length are used to crop the original data. Both training dataset and testing dataset are cropped to keep their length consistent in the time dimension. For training dataset, all the cropped data are reserved in order to increase the size of the dataset. For testing dataset, only the last 3.9s data that containing intact motor imagery signals are reserved.

2) Adding Noise: By adding Gaussian noise to the original data, new training data are generated while retaining the features of the original data. The probability density function of Gaussian noise obeys Gaussian distribution:

$$p_G(z) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(z-\mu)^2}{2\sigma^2}} \tag{4}$$

TABLE II
CLASSIFICATION USING DIFFERENT RATIOS OF EXPANDED SYNTHESIZED DATA ON THE FIRST SUBJECT (MEAN±STD, %)

Number of real samples	Ratio						
	Real-only	0.5	1	1.5	2	3	4
40	31.4±3.5	33.7±2.6	32.3±4.1	34.1±3.7	34.4±3.5	**34.7±3.0**	33.8±3.2
80	35.8±4.6	36.4±4.0	38.1±5.2	39.1±3.9	39.6±4.6	**40.7±5.1**	38.8±4.0
120	44.1±6.2	43.3±6.1	**50.7±5.1**	46.4±4.5	50.4±5.9	47.8±2.8	48.7±5.6
160	48.9±8.5	47.1±9.2	56.7±4.8	52.4±6.3	**58.4±2.9**	57.8±4.2	56.3±3.2
200	47.1±6.2	50.7±6.2	53.6±5.7	51.9±5.6	56.3±2.9	**58.3±4.4**	56.4±4.3
240	52.4±6.5	52.1±8.1	57.7±3.9	60.4±2.2	**61.7±3.2**	60.3±2.3	60.3±4.1
280	55.9±6.6	58.8±4.2	58.5±2.8	60.4±3.5	62.3±1.6	**62.8±3.5**	62.2±2.6
320	61.7±4.6	61.4±2.4	63.4±3.8	64.8±4.0	67.5±2.5	65.4±2.7	**68.3±2.9**
360	61.6±5.5	64.1±3.6	65.2±3.9	66.8±4.2	67.6±3.4	67.7±4.4	**68.1±2.5**
400	64.6±3.9	65.3±2.0	67.6±2.6	67.7±3.4	69.1±2.5	69.6±3.3	**71.6±3.7**
440	65.4±3.3	67.9±4.0	67.8±2.4	68.3±2.6	68.5±2.4	67.3±2.0	**69.4±4.0**
480	68.3±3.8	69.4±3.9	69.9±5.0	69.6±4.5	72.2±3.4	**73.3±5.6**	69.1±4.4
520	65.9±5.7	66.4±5.6	72.3±6.4	70.4±4.0	75.4±6.1	**76.0±4.8**	74.0±3.9
Average	54.1±5.3	55.1±4.8	58.0±4.3	57.9±4.0	**60.3±3.5**	60.1±3.7	59.8±3.7
Improvement	0	+1	+3.9	+3.8	**+6.2**	+6	+5.7

The average of 'Real-only' is used as a benchmark to calculate the improvement after applying the proposed method. '0' represents o improvement, '+' means positive improvement and '-' means negative improvement, the meaning is applicable to the other tables. **Bold** represents the highest accuracy obtained by the current number of samples

TABLE III
WITHIN-SUBJECT CLASSIFICATION FOR ALL SUBJECTS BEFORE AND AFTER USING THE PROPOSED METHOD (MEAN±STD, %)

Subject	Number of real samples									
	Real-only		0.5		1		1.5		2	
	Real-only	This work	Real-only	This work	Real-only	This work	Real-only	This work	Real-only	This work
1	31.4±3.5	34.4±3.5	48.9±8.5	58.4±2.9	55.9±6.6	62.3±1.6	64.6±3.9	69.1±2.5	65.9±5.7	75.4±6.1
2	30.2±3.8	33.9±1.9	41.9±5.9	45.9±4.7	49.7±4.8	54.2±2.1	53.4±4.1	59.3±3.3	59.6±5.9	63.6±4.2
3	36.5±4.4	39.3±3.6	55.8±3.7	59.7±4.1	66.9±3.6	69.9±2.8	74.2±6.7	78.9±1.9	74.5±2.4	82.7±4.0
4	27.5±2.1	29.6±1.5	40.2±5.9	42.9±4.9	52.4±6.8	54.5±3.6	57.6±3.1	64.6±5.0	66.4±8.3	75.3±4.1
5	31.2±3.7	35.1±3.6	58.1±6.7	63.0±3.4	70.7±3.0	74.1±2.1	75.1±3.4	78.7±1.2	76.1±4.9	79.6±4.1
6	28.7±2.3	31.3±2.9	43.4±5.3	46.5±3.1	52.4±4.4	54.9±1.9	53.8±4.5	58.2±3.8	61.3±6.7	61.8±2.3
7	35.3±4.4	38.7±5.8	61.6±8.1	67.4±2.2	72.1±1.9	72.5±1.3	73.6±2.5	76.3±2.1	72.2±2.6	75.3±2.6
8	39.2±4.7	43.6±4.7	58.7±6.2	63.0±5.0	67.5±4.1	72.9±2.0	70.5±4.9	79.4±3.1	82.0±3.4	86.6±3.6
9	32.1±4.8	39.3±7.7	55.7±8.0	61.2±3.2	70.5±6.0	76.4±2.0	68.8±2.6	82.3±2.2	82.6±3.4	88.0±2.6
Average	32.5±3.7	36.1±3.9	51.6±6.5	56.4±3.7	62.0±4.6	65.7±2.2	65.7±4.0	71.9±2.8	71.2±4.8	76.5±3.7
Improvement	0	+3.6	0	+4.8	0	+3.7	0	+6.2	0	+5.3

where z is the random variable, μ is the mean, σ is the standard deviation. We set $\mu = 0$, $\sigma = 0.1$ in this study, and only add noise to the training dataset.

3) GAN: GAN [25] consists of generator and discriminator. Traditional GAN needs to train multiple generators to generate multiple types of samples, but its variant cGAN [57] can impose constraints on generator and discriminator to synthesize specified samples. In this paper, cGAN is used to directly synthesize four types of samples. The structure of the generator is consistent with that mentioned in Section II-B. We redesign the discriminator to form a confrontation between the two models, which consists of four convolutional layers and two fully connected layers, and the structure is shown in Table I.

4) Decoding Models Used for Replacement: EEGNet [47], ShallowConvNet [29], DeepConvNet [29] are used as auxiliary decoding model to test the framework. We modify the size of temporal convolution kernel and pooling kernel in EEGNet to twice the original size according to author's suggestion. The size of spatial convolution kernel is set to 22 for the three decoders, and the number of hidden units in fully connected layer is modified according to input size, other parameters are the same as the original and can be found in [29] and [47].

D. Cross-Validation Analysis

As shown in Fig. 3, all real samples are split into training set, verification set and test set for ten times. The total number of samples in training set and verification set is equal to the number of real samples that set in each experiment (i.e., 40, 80, …, 520). Training samples account for 90% and verification samples account for 10%. The test set samples are all real samples except the training and verification samples.

After each dataset splitting, the training and verification set are input into the synthesis framework to generate synthesized samples. Then the training samples, synthesized samples and verification samples are used to train and verify the classifier. Real test samples are used to test and evaluate the classifier.

E. Evaluation Metrics

In visualization section, data are mainly evaluated by visual inspection. Accuracy and standard deviation are used to

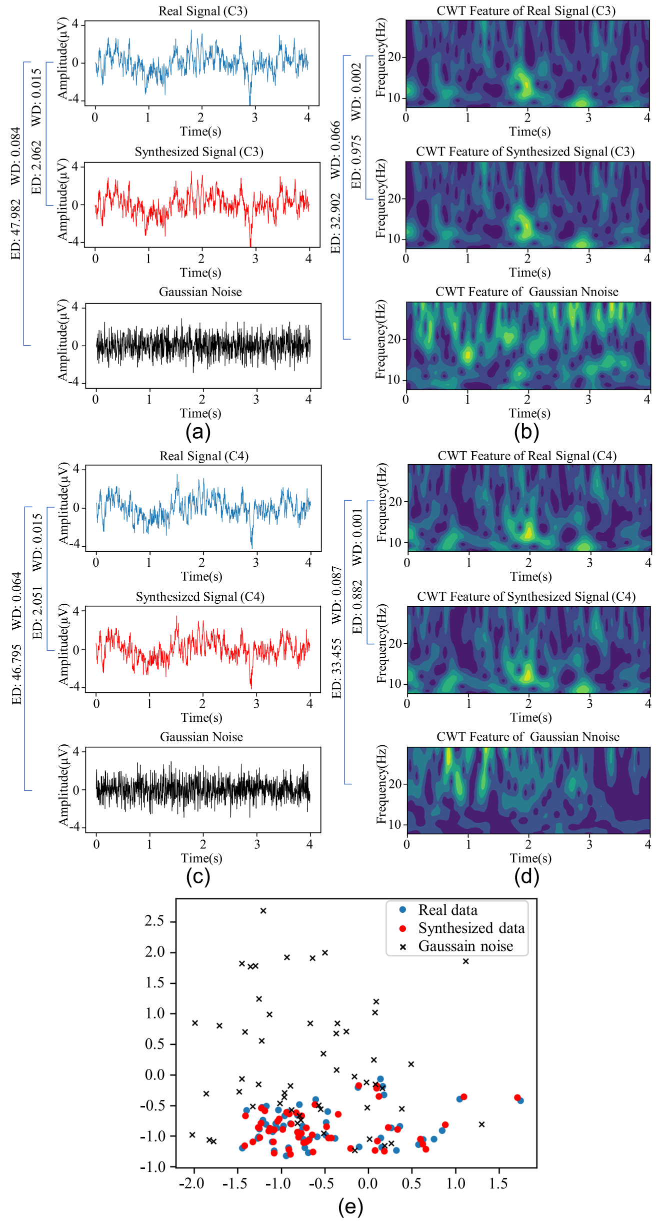

Fig. 4. Visualization of real signals, synthesized signals and signals generated by Gaussian noise. (a) Waveform of signals on C3 channel, (b) Time-frequency features extracted by CWT on C3 channel, (c) Waveform of signals on C4 channel, (d) Time-frequency features extracted by CWT on C4 channel, (e) Spatial features extracted by CSP. ED, WD are used for evaluation, FID is not applicable to evaluate single sample.

evaluate the classification performance of the model. Following metrics are also used for performance evaluation.

1) Fréchet Inception Distance (FID): FID is commonly used to evaluate the quality of generative model and synthesized samples [58]. Compared with Inception Score (IS), this metric is more robust to noise and more sensitive to the quality of the generative model. FID uses a pre-trained classifier to compare the feature distribution of real samples and synthesized samples in the embedded layer.

2) Wasserstein Distance (WD): WD describes the cost of converting one distribution to another under a given cost function [59], and is often used to measure the similarity between any two distributions.

3) Euclidean Distance (ED): ED is used to evaluate the similarity between samples. The minimum Euclidean Distance (ED_{min}) calculate the minimum distance between real samples or the minimum distance between real samples and synthesized samples. The ED_{min} between real samples and synthesized samples should be equivalent to the minimum distance distribution between real samples [60].

III. Results

A. Visualization of Synthesized Data

The synthesized data are visualized from the time, frequency, and spatial domains. Gaussian noise with the zero mean and unit variance is used for comparison with the synthesized data. Gaussian noise is one of the inputs ofthe generative model, and the statistical distribution of it is the same as the normalized EEG signal, which also with zero mean and unit variance.

The waveform of the data is directly evaluated by visual inspection in time domain analysis. In frequency analysis, we use continuous wavelet transform (CWT) [61] to transform the data. Since motor imagery leads to energy changes in alpha-band (8-13Hz) and beta-band (13-30Hz) [62], the time-frequency features within the range of 8-30Hz are selected to analyzed. For spatial analysis, common spatial pattern (CSP) [63] is used to extract two-dimensional spatial features of the data.

Fig. 4 shows the waveform, time-frequency features and spatial features of the real data, synthesized data and Gaussian noise in C3 and C4 channel. These two channels are typically related to MI features [17]. According to visual inspection and evaluation metrics of WD and ED, the distributions of these three features of the synthesized data are similar to that of real data while there is a significant difference between Gaussian noise and real data. The similarity means that the synthesized data effectively preserves the time-frequency-spatial features of the real data.

B. Expansion Ratio Explore and Overall Performance

The performance of the deep learning model depends on the number of training data, thus adding different ratio of synthesized data to the training set will have different effects on the classification performance. To find an appropriate expansion ratio, we select different numbers of real samples for cross-validation on the first subject, and the expansion ratios of training dataset are set to 0.5, 1, 1.5, 2, 3 and 4. As shown in Table II. The classification accuracy of the model after expansion is better than that without expansion (p-value<0.05 for 0.5 expansion ratio and p-value<0.001 for other ratios. Wilcoxon signed-rank test is used for assessment and Holm-Bonferroni approach for correction), and the standard deviation decreases after data augmentation, which means that the model is more stable. When the number of expanded samples is twice the number of original training samples, the average accuracy is the highest, which is 6.2% higher than the average accuracy without expansion. In addition,

TABLE IV
CLASSIFICATION AFTER USING DIFFERENT AUGMENTATION METHODS ON THE FIRST SUBJECT (MEAN±STD, %)

Number of real samples	Real-only	This work	Cropping	Adding noise	GAN
40	31.4±3.5	33.7±2.6	32.3±4.1	34.1±3.7	/
80	35.8±4.6	39.6±4.6	37.4±5.9	37.6±2.5	/
120	44.1±6.2	50.4±5.9	48.9±9.4	42.4±5.0	46.8±4.1
160	48.9±8.5	58.4±2.9	53.7±7.0	48.7±6.3	51.9±9.6
200	47.1±6.2	56.3±2.9	55.8±2.2	49.8±3.2	51.5±4.1
240	52.4±6.5	61.7±3.2	58.5±3.1	53.9±4.5	61.2±8.7
280	55.9±6.6	62.3±1.6	59.1±3.4	58.8±5.1	59.7±6.1
320	61.7±4.6	67.5±2.5	63.5±4.5	62.2±3.8	65.7±3.9
360	61.6±5.5	67.6±3.4	63.9±2.3	65.2±2.9	64.1±4.0
400	64.6±3.9	69.1±2.5	65.0±2.4	65.7±5.2	66.8±5.0
440	65.4±3.3	68.5±2.4	65.2±2.8	67.3±3.8	69.3±4.2
480	68.3±3.8	72.2±3.4	67.4±4.3	69.7±4.4	68.0±2.2
520	65.9±5.7	75.4±6.1	67.7±6.2	68.9±4.1	69.4±2.5
Average	54.1±5.3	60.3±3.5	56.8+4.3	55.6+4.1	57.0+4.2
Improvement	0	+6.2	+2.7	+1.5	+3

'/' represents cannot get synthesized data, and it is instead by the accuracy of Real-only when calculating the average accuracy of GAN.

TABLE V
EVALUATION SCORES OF 520 SAMPLES SYNTHESIZED BY DIFFERENT AUGMENTATION METHODS FOR THE FIRST SUBJECT. REAL AND NOISE ARE USED AS REFERENCE (MEAN±STD, %)

Method	FID	WD	ED$_{min}$
This work	**0.059**	0.034	**143.300**
Cropping	2.073	**0.016**	/
Adding noise	0.497	0.047	152.551
GAN	8.340	0.203	132.532
Real	0.	0.	145.610
Noise	188.350	0.115	182.091

'/' represents the score cannot be obtain because the length of the data is different between real and synthesized samples. **Bold** represents the best score.

TABLE VI
CLASSIFICATION AFTER REPLACING THE AUXILIARY DECODING MODEL WITH OTHER STATE-OF-THE-ART DECODING MODELS ON THE FIRST SUBJECT (MEAN±STD, %)

Number of real samples	EEGNet		ShallowConvNet		DeepConvNet	
	Real-only	This work	Real-only	This work	Real-only	This work
40	31.4±3.5	33.7±2.6	32.3±4.1	34.1±3.7	34.1±3.7	34.1±3.7
80	48.9±7.8	50.7±7.2	52.7±2.4	54.6±1.9	31.4±2.5	33.2±2.6
120	56.3±10.5	62.3±5.1	56.8±2.6	59.4±2.3	32.7±3.5	41.4±3.2
160	63.9±7.8	68.2±4.3	62.8±2.4	63.2±2.0	33.7±3.5	48.5±3.7
200	66.8±2.3	69.3±3.0	61.1±2.2	64.7±2.3	42.2±8.5	52.4±5.1
240	69.2±4.1	73.9±3.5	65.9±2.2	67.6±1.5	50.6±7.2	55.9±4.9
280	70.8±2.6	74.7±2.0	67.3±3.4	69.1±2.6	50.8±4.2	59.5±4.1
320	76.0±3.0	77.0±4.4	71.3±2.4	72.1±2.7	55.8±6.0	61.1±3.0
360	75.3±3.8	81.0±2.2	72.1±1.6	73.6±1.4	57.5±6.1	65.6±4.2
400	79.1±2.7	80.6±2.3	73.7±2.8	75.6±3.0	59.3±4.9	65.8±5.5
440	77.9±3.5	80.8±1.9	73.9±3.2	74.7±2.4	61.8±5.1	70.2±3.3
480	80.8±3.0	80.5±2.9	73.0±2.5	72.8±2.6	63.0±4.0	68.7±2.2
520	79.6±4.3	85.6±3.7	73.7±4.6	76.9±2.8	66.5±6.9	75.6±4.0
Average	67.7±4.6	71.1±3.5	64.7±2.9	66.8±2.3	49.0±4.9	56.0±3.6
Improvement	0	+3.4	0	+2.1	0	+7

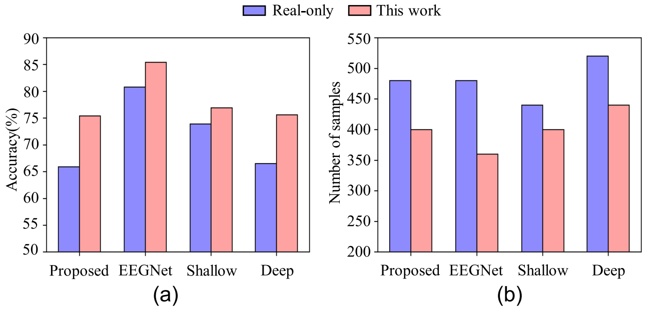

Fig. 5. Comparison of accuracy and number of training samples before and after applying the proposed method. (a) The highest accuracy of each decoder, (b) The number of samples required for each decoder to achieve the highest accuracy with only real data, and the minimum number of samples required to achieve the same or higher accuracy after applying the proposed method. Proposed: Our decoder, Shallow: ShallowConvNet, Deep: DeepConvNet.

at least 480 samples are required for 68.3% accuracy without expansion, while only 320 samples are required to achieve this accuracy when the ratio of expanded synthesized data is 4, a decrease of 33.3%.

The original decoding performance and the decoding performance after applying proposed method are tested using the expansion ratio of 2 for all subjects, the result is shown in Table III. After applying the proposed method, the average accuracy of all subjects under different number of real samples is improved by (4.72±0.98)%, and the maximum improvement of 6.2% is obtained when the number of real samples is 400.

C. Comparison of Different Augmentation Methods

We compare the performance of three different data augmentation methods under different numbers of real samples, including proposed method, cropping and adding noise. The expansion ratio is set to 2. As shown in Table IV, the proposed method significantly improves the model performance under different conditions (p-value<0.001). Compared with the result using only real data, the average accuracy is improved by 6.2%, which is higher than 2.7%, 1.5% and 3% for cropping, adding noise and GAN. Further research reveals an interesting phenomenon that cropping is more suitable for the case of insufficient data, adding noise and GAN are more suitable for the case of relatively sufficient data, while our method can simultaneously take into account different conditions.

The quality of synthesized samples of different augmentation methods is evaluated by FID, ED$_{min}$ and WD. We analyze the synthesized samples obtained when the number of real samples is 520.The distance between the real samples, and the distance between the real samples and the noise samples are calculated as the reference. Table V shows that the FID and ED$_{min}$ of samples synthesized by the proposed method is the closest to the real samples, which is superior to other methods. The proposed method is inferior to Cropping in WD.

D. Generality of Auxiliary Synthesis Framework

To prove that the proposed framework is also applicable to other decoding models, we replace the auxiliary decoding model with EEGNet, ShallowConvNet and DeepConvNet, respectively, and then test the model performance, the result can be found in Table VI. The data augmentation method proposed in this paper improves the average accuracy of these

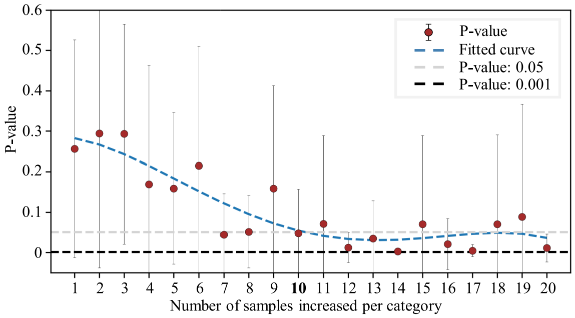

Fig. 6. Statistical difference changes of recognition results when the number of training samples is increased at different step sizes.

three models by 3.4%, 2.1% and 7%, respectively, and the stability of the model is also increased. DeepConvNet obtains the greatest improvement, the result reveals that our method may be more compatible with complex models.

Table II-VI also reveal that after applying the proposed method, the highest accuracy of all models is higher than that of training only with the real data. Fig. 5(a) shows the intuitively histogram of this result. Besides, we count the number of samples required to achieve the highest accuracy when training the model only with real data, and the minimum number of samples required to achieve the same or higher accuracy after augmentation, as shown in Fig. 5(b). The number of samples required to train the decoding model to achieve the same or higher accuracy decreases after applying our method.

IV. Discussion

A. Setting of Step Size and Expansion Ratio

The selection of step size requires to consider the sample balance, the significance of the results and the test error. The most important thing is to ensure the sample balance, which has been proved to affect the model training effect, so each kind of sample needs to be balanced in the process of increasing. And then, in order to avoid the time consumption caused by unnecessary tests, the step size that will lead to significant changes in the results is selected in the study. We test the performance of the model with different step sizes based on real samples. The accuracy of different number of real samples is used as reference. On the basis of these samples, each type of sample is increased with different step sizes, and the statistical difference between the obtained accuracy and the reference is tested by Wilcoxon signed-rank test, and corrected by Holm-Bonferroni function. As shown in Fig. 6, with the increase of step size, the p-value decreases, and the step size of 10 is close to the significant level. Although the larger the step size, the more significant the change, choosing a larger step size will cause greater calculation error of sample reduction. When the same accuracy is achieved, too large a step size may lead to a much larger number of samples than the real demand when using data augmentation method to achieve this accuracy, which will result in a smaller sample reduction.

The number of synthesized samples in the training set affects the improvement of model performance. Table VII shows the data expansion ratio commonly used in EEG recognition literature in the BCI field, in which the optimal ratio refers to the expansion ratio that makes the model performance reach the optimal after expansion. According to the relevant references, the conventional expansion ratios of 1, 2, 3 and 4 are set in this study, while the ratios of 0.5 and 1.5 are set to test the performance change of the model at a smaller expansion ratio. A larger ratio is not used because the improvement of model performance by synthesized samples tends to be saturated when the ratio larger than 2.

TABLE VII
Expansion Ratio Used in Recent Studies

Study	Method	Expansion ratio	Best ratio
Luo [14]	cWGAN	0, 1, 2, 3, 4, 5	1
Zhang [64]	Empirical mode decomposition	0, 1, 2, 3, 4, 5	1
Zhang [13]	cDCGAN	0, 0.5, 1, 1.5, 2	2
Xu [43]	GAN	0, 3, 5, 10	10
Fahimi [65]	cDCGAN	1	1
This work	Auxiliary generation	0, 0.5, 1, 1.5, 2, 3, 4	2

TABLE VIII
Training Time of Different Methods

Method	Stage 1	Stage 2	Total
This work	31s	179.1s	210.1s
GAN	/	259.1s	259.1s

All models are trained for 100 epochs with 468 real samples. Stage 1 means the pre-training of auxiliary decoding model, stage 2 means the training of generative model.

B. Complexity of Designing and Training Progress

In this study, we propose a data synthesis framework based on deep generative model. The framework only needs to design a decoding model and a generative model, and because there is no adversarial relationship between the two models, we do not need to make complicated parameter tuning. In contrast, GAN-based data augmentation method realizes data synthesis through adversarial training, which is extremely sensitive to the hyperparameters. In this paper, we try to use the decoding model mentioned in Section II-B as discriminator and find that the discriminator is always unable to distinguish real and fake, so that the GAN cannot converge. After redesign, this phenomenon is alleviated. The difference in the early network structure design indicates that the proposed framework is much easier to implement.

The training process of this framework is stable, and the quality of the generative model can be judged by the loss. Furthermore, the result in Table IV shows that the minimum data required for the training of this framework is less than that of GAN, because when the number of the real samples is less than 120, effective synthesized data cannot be obtained by GAN. Training time of this framework is also less than that of GAN, as presented in Table VIII.

C. Ablation Experiment

In order to verify the effectiveness of the proposed method and verify that the synthesized sample is not a copy of

TABLE IX
CLASSIFICATION UNDER DIFFERENT ABLATION CONDITIONS ON THE FIRST SUBJECT (MEAN±STD, %)

Number of real samples	MSE-CE-based (This work)	MSE-based	CE-based
40	34.4±3.5	33.3±2.5	29.0±2.3
80	39.6±4.6	38.5±4.1	32.2±3.2
120	50.4±5.9	49.4±5.4	42.5±7.2
160	58.4±2.9	54.7±7.1	46.0±9.2
200	56.3±2.9	53.3±7.1	46.4±6.3
240	61.7±3.2	57.8±3.1	52.2±5.8
280	62.3±1.6	60.5±4.7	56.3±3.4
320	67.5±2.5	64.0±1.9	63.1±2.9
360	67.6±3.4	64.6±4.6	59.5±5.2
400	69.1±2.5	70.2±4.9	64.7±4.0
440	68.5±2.4	66.8±3.9	62.9±2.8
480	72.2±3.4	69.3±3.3	67.0±3.4
520	75.4±6.1	72.6±5.0	64.3±5.7
Average	60.3±3.5	58.1±4.4	52.8±4.7
Improvement	0	-2.2	-7.5

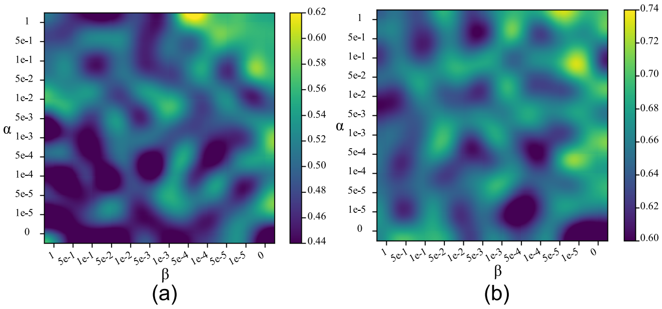

Fig. 7. Interaction between α and β in the auxiliary synthesis framework. The hyperparameter varies by five times. Color bar represents the corresponding accuracy. (a) 200 real samples, (b) 400 real samples.

the real sample but added information which is helpful to improve the model performance, we tested the influence of each component in the synthesis framework on the synthesis result and decoding performance from different perspectives through ablation experiments. The ablation experiments are set as follows:

MSE-CE-Based (This Work): The loss functions involved in optimization include MSE loss and CE loss, set α to 1, β to 0.0001.

MSE-Based: The loss function involved in optimization only includes MSE loss, and the auxiliary decoding model in the synthesis framework is removed.

CE-Based: The loss function involved in optimization only includes CE loss.

We test the decoding model using the samples synthesized under these three conditions. The average accuracy of MSE-CE-based condition is 2.3% and 7.2% higher than that of MSE-based and CE-based condition, and the model is more stable, as shown in Table IX. When the number of real samples is less than 400, the samples synthesized with the participation of the auxiliary decoding model can stably improve the decoding accuracy, but when the number of real samples is more than 400, the impact of the auxiliary decoding model on the decoding accuracy may be unstable, and it is necessary to reduce the constraints of the auxiliary decoding model on the generative model.

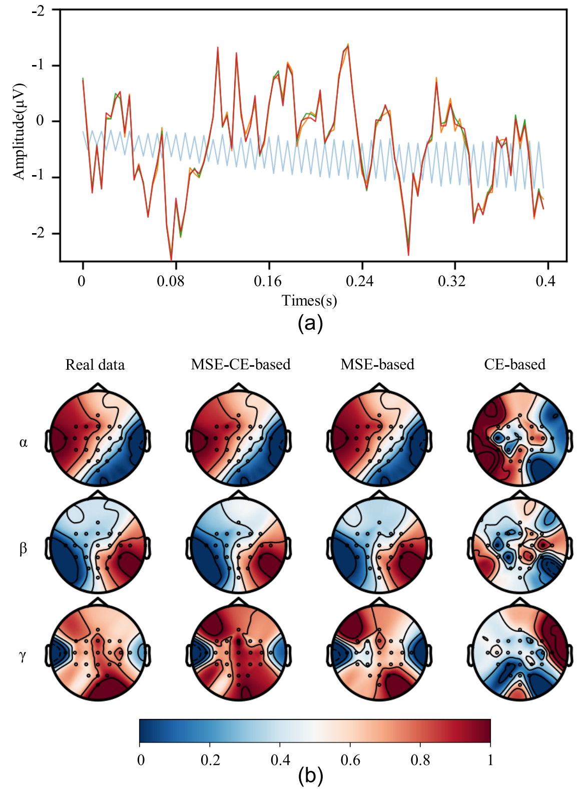

Fig. 8. Time and frequency evaluation of real signal and signals synthesized under MSE-CE-based, MSE-based and CE-based conditions. (a) The 0-0.4s signals of C3 channel of one trial are selected for a detailed inspection, (b) Brain topographic maps when subject imagines 'left hand' movement. Alpha, beta, gamma bands are chosen for comparison.

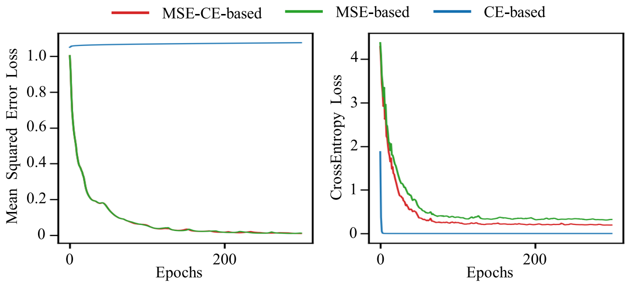

Fig. 9. Losses of generative model under different conditions. For MSE-based condition, the CE loss is only calculated for comparison, and it does not participate in optimization. For CE -based condition, the MSE loss is only calculated for comparison, and it does not participate in optimization.

Grid search ranging from 1 to 1e-5 is used to find an appropriate coefficient combination, we also test the situation that the coefficient is zero. Fig. 7 shows the two coefficients and the corresponding accuracy after Gaussian interpolation when there are 200 and 400 samples. The accuracy is generally higher than other combinations when α is larger than 1e-1 and β is smaller than 5e-4. The β required for the highest accuracy decreases from 5e-4 to 1e-5 when the number of samples increases from 200 to 400.

To further investigate the reasons for the disparity of ablation experiment, we visualized the effect of different conditions on the generative model, and the effect of synthesized

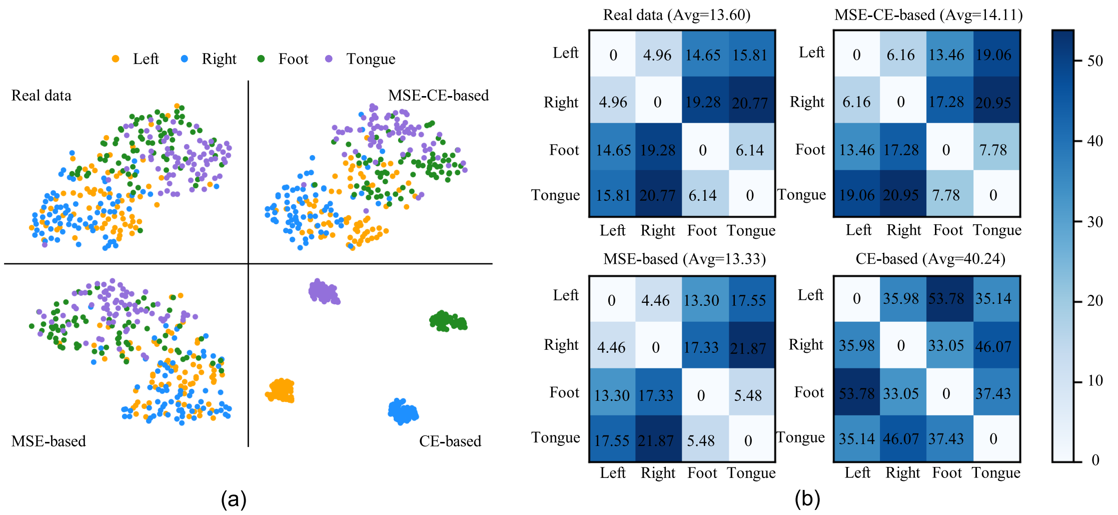

Fig. 10. t-SNE visualization of the decoding model outputs when training it with only real data or MSE-CE-based, MSE-based and CE-based synthesized data. (a) The t-SNE visualization for the outputs of the last convolution layer, (b) Euclidean distance between different clusters in (a).

samples under different conditions on the training and testing of the decoding model.

1) Influence on the Results of Generative Model: To specifically explain the impact of different components in the synthesis framework on the synthesized data, we observe the outputs and the loss curves of the generative model.

Fig. 8 shows the time and frequency evaluation of the real data and synthesized data under different conditions. The alpha and beta bands associated with motor imagery and the gamma band associated with complex tasks and cognition are selected for brain topographical map comparison. The results show that except for CE-based condition, the synthesized data under the other two conditions have similar distributions in alpha and beta bands to the real data while the distribution of the gamma band is significantly different. As mentioned in [60], the generated signal is expected to have additional high-frequency features, which have a positive effect on improving the generalization of the model.

We record the changes of these two losses in the training process of generative model. Fig. 9 reveals that the MSE loss of the synthesized signal is similar under the condition of MSE-CE-based and MSE-based, while is extremely large under CE-based, indicating that the synthesized data distribution is deviated from the real distribution when the MSE loss does not participate in optimization. The CE loss under MSE-CE-based condition is smaller than that of MSE-based condition, which means that the CE loss provided by pre-trained decoding mode helps the generative model synthesize distinguishing features in the synthesis process. Fig. 9 also reveals that there is a big difference between the values of MSE and CE loss in the training stage, which proves that a larger α and a smaller β are more conducive to balancing the constraints of the two losses.

2) Influence on the Training of Decoding Model: We further study the influence of the samples synthesized under different conditions on the decoding model. The convolution outputs of the decoding model are analyzed by t-SNE [66] from the perspective of training. In this experiment, only the synthesized samples are used to train the decoding model.

Fig. 10(a) shows that the potential vectors of the samples synthesized under the CE-based condition are obviously distinctive between different motor imagery, which proves that CE loss assists the generator in synthesizing distinctive features. But as shown in Fig. 8 and Fig. 9, only using the CE loss will cause the data distribution of the synthesized samples to deviate from the real EEG signals. These samples can be easily classified by the decoding model, but the model trained with these samples cannot accurately recognize the real samples during the testing. Fig. 11 shows that the model trained with the samples synthesized under the CE-based condition has differences in the classification information focus area of the real samples during the testing compared with other conditions, which leads to the results of the CE-based condition in Table IX being inferior to other ablation conditions.

An appropriate using of the auxiliary decoding model will make the feature distributions of different motor imagery are more distinct than real only and MSE-based condition while retaining the real signal distributions. The Euclidean distance between different clusters is shown in Fig. 10(b). After adding the auxiliary decoding model, the average distance between clusters increases, especially the distance between left and right hand and the distance between foot and tongue are significantly increased, which means that the model is more capable of identifying these movements at this time.

3) Influence on the Testing of Decoding Model: The outputs of the decoding model are further analyzed from the perspective of testing. In this experiment, we use both real samples and synthesized samples obtained under different conditions to train the decoding model, and then select the same samples for testing. Class activation mapping (CAM) [67] is used to visualize the performance of the decoding model on the testing dataset, as shown in Fig. 11. The energy distribution of the CAM is different when the same sample is correctly classified by the decoding models trained under the different conditions. According to the statistical analysis of random samples, MSE-CE-based condition has larger energy and narrower focused area, which means that the decoding model captures the

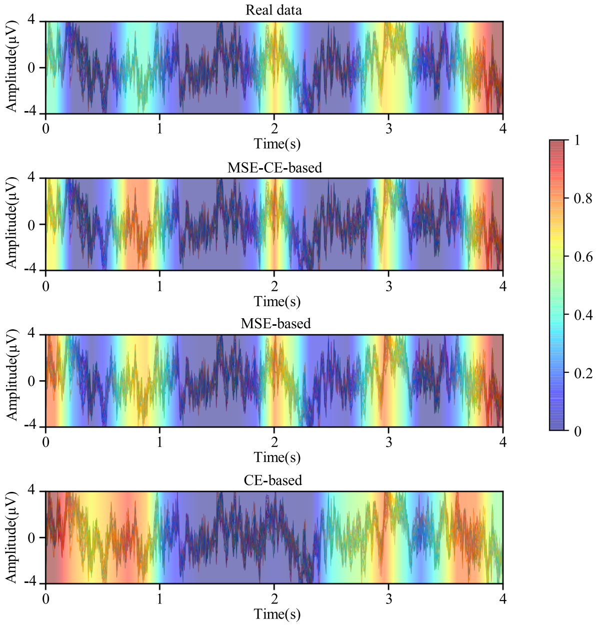

Fig. 11. Grad-CAM visualizations under MSE-CE-based, MSE-based and CE-based conditions. Color bar represents the impact on classification results. The figure shows the class activation mapping when the decoding models trained under the different conditions correctly classify the same sample.

information needed for classification more precisely, this may improve the classification performance.

D. Limitation and Future Work

The proposed method improves the decoding performance with limited real samples. However, the generative model learns the real data distribution to generate artificial samples and cannot generate information that is not included in the original dataset. The performance of the augmentation methods is affected by the data quality and diversity of the real dataset. Therefore, the improvement of decoding model performance by synthesized samples is limited. In the actual acquisition experiment of EEG signals, some random changes, such as physiological individual differences and interference caused by the environment, cannot be simulated by deep learning method at present. Besides, the overfitting still exists, so the early stop strategy is used to prevent the performance of the model from decreasing. We expect to achieve acceptable decoding accuracy with few training samples, or even without retraining. In the follow-up research, we will continue to study how to train a model with very few samples to achieve acceptable accuracy, such as combining our method with transfer learning

V. Conclusion

In this paper, an auxiliary synthesis framework is proposed to effectively improve the classification performance of the model under limited samples. We tested the method on BCI Competition IV 2a, and the results show that the data synthesized by this method well preserves the time, frequency and spatial features of the original data. After applying the proposed method, the average decoding accuracy of all subjects is improved by (4.72±0.98)%. A detailed investigations on the first subject shows the improvement of classification performance brought by our method is higher than that of the cropping, adding noise and GAN, and also higher than that without the auxiliary decoding model in the framework. The number of training samples for reaching the original highest accuracy is reduced by about 33.3%. The proposed method is also applicable to other decoding models. Finally, since our framework is purely data-driven, it can be migrated into other BCI domains such as emotion recognition, speech recognition, epilepsy prediction etc.

References

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

[2] F. Cincotti et al., "High-resolution EEG techniques for brain–computer interface applications," *J. Neurosci. Methods*, vol. 167, no. 1, pp. 31–42, Jan. 2008.

[3] E. P. P. Torres, E. A. H. Torres, M. Hernández-Álvarez, and S. G. Yoo, "EEG-based BCI emotion recognition: A survey," *Sensors*, vol. 20, no. 18, p. 5083, Sep. 2020.

[4] D. A. Handelman et al., "Shared control of bimanual robotic limbs with a brain-machine interface for self-feeding," *Frontiers Neurorobot.*, vol. 16, Jun. 2022, Art. no. 918001.

[5] Z. Liang et al., "EEGFuseNet: Hybrid unsupervised deep feature characterization and fusion for high-dimensional EEG with an application to emotion recognition," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 1913–1925, 2021.

[6] G. K. Anumanchipalli, J. Chartier, and E. F. Chang, "Speech synthesis from neural decoding of spoken sentences," *Nature*, vol. 568, no. 7753, pp. 493–498, Apr. 2019.

[7] H. Khan, L. Marcuse, M. Fields, K. Swann, and B. Yener, "Focal onset seizure prediction using convolutional networks," *IEEE Trans. Biomed. Eng.*, vol. 65, no. 9, pp. 2109–2118, Sep. 2018.

[8] M. Sokolovsky, F. Guerrero, S. Paisarnsrisomsuk, C. Ruiz, and S. A. Alvarez, "Deep learning for automated feature discovery and classification of sleep stages," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 17, no. 6, pp. 1835–1845, Nov. 2020.

[9] X. Zhang, L. Yao, X. Wang, J. Monaghan, D. McAlpine, and Y. Zhang, "A survey on deep learning-based non-invasive brain signals: Recent advances and new frontiers," *J. Neural Eng.*, vol. 18, no. 3, Mar. 2021, Art. no. 031002.

[10] J. Xie et al., "A transformer-based approach combining deep learning network and spatial–temporal information for raw EEG classification," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 2126–2136, 2022.

[11] Z. Liu, F. He, J. Tang, B. Wan, and D. Ming, "Research advancements of deep learning on EEG decoding," *Chin. J. Bio-Med. Eng.*, vol. 39, no. 2, pp. 215–228, Apr. 2020.

[12] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Generalizing from a few examples: A survey on few-shot learning," *ACM Comput. Surv.*, vol. 53, no. 3, pp. 1–34, May 2021.

[13] Q. Zhang and Y. Liu, "Improving brain computer interface performance by data augmentation with conditional deep convolutional generative adversarial networks," 2018, *arXiv:1806.07108*.

[14] Y. Luo and B.-L. Lu, "EEG data augmentation for emotion recognition using a conditional Wasserstein GAN," in *Proc. 40th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2018, pp. 2535–2538.

[15] Y. Luo, L.-Z. Zhu, Z.-Y. Wan, and B.-L. Lu, "Data augmentation for enhancing EEG-based emotion recognition with deep generative models," *J. Neural Eng.*, vol. 17, no. 5, Oct. 2020, Art. no. 056021.

[16] D. Tkach, J. Reimer, and N. G. Hatsopoulos, "Observation-based learning for brain–machine interfaces," *Current Opinion Neurobiol.*, vol. 18, no. 6, pp. 589–594, Dec. 2008.

[17] S. K. R. Singanamalla and C.-T. Lin, "Spiking neural network for augmenting electroencephalographic data for brain computer interfaces," *Frontiers Neurosci.*, vol. 15, Apr. 2021, Art. no. 651762.

[18] 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.

[19] M. M. Krell and S. K. Kim, "Rotational data augmentation for electroencephalographic data," in *Proc. 39th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2017, pp. 471–474.

[20] E. Lashgari, D. Liang, and U. Maoz, "Data augmentation for deep-learning-based electroencephalography," *J. Neurosci. Methods*, vol. 346, Dec. 2020, Art. no. 108885.

[21] J. Kevric and A. Subasi, "Comparison of signal decomposition methods in classification of EEG signals for motor-imagery BCI system," *Biomed. Signal Process.*, vol. 31, pp. 398–406, Jan. 2017.

[22] Q. Wen et al., "Time series data augmentation for deep learning: A survey," 2020, *arXiv:2002.12478*.

[23] M. Xu, Y. Chen, Y. Wang, D. Wang, Z. Liu, and L. Zhang, "BWGAN-GP: An EEG data generation method for class imbalance problem in RSVP tasks," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 251–263, 2022.

[24] C. He, J. Liu, Y. Zhu, and W. Du, "Data augmentation for deep neural networks model in EEG classification task: A review," *Frontiers Hum. Neurosci.*, vol. 15, Dec. 2021, Art. no. 765525.

[25] I. J. Goodfellow et al., "Generative adversarial networks," 2014, *arXiv:1406.2661*.

[26] A. Al-Saegh, S. A. Dawwd, and J. M. Abdul-Jabbar, "CutCat: An augmentation method for EEG classification," *Neural Netw.*, vol. 141, pp. 433–443, Sep. 2021.

[27] E. Lashgari, J. Ott, A. Connelly, P. Baldi, and U. Maoz, "An end-to-end CNN with attentional mechanism applied to raw EEG in a BCI classification task," *J. Neural Eng.*, vol. 18, no. 4, Aug. 2021, Art. no. 0460e3.

[28] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, no. 1, pp. 321–357, Jan. 2002.

[29] 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, Dec. 2017.

[30] 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.

[31] Z. Mousavi, T. Y. Rezaii, S. Sheykhivand, A. Farzamnia, and S. N. Razavi, "Deep convolutional neural network for classification of sleep stages from single-channel EEG signals," *J. Neurosci. Methods*, vol. 324, Aug. 2019, Art. no. 108312.

[32] T.-J. Luo, C.-L. Zhou, and F. Chao, "Exploring spatial-frequency-sequential relationships for motor imagery classification with recurrent neural network," *BMC Bioinf.*, vol. 19, no. 1, p. 344, Sep. 2018.

[33] Z. Tayeb et al., "Validating deep neural networks for online decoding of motor imagery movements from EEG signals," *Sensors*, vol. 19, no. 1, p. 210, Jan. 2019.

[34] I. Majidov and T. Whangbo, "Efficient classification of motor imagery electroencephalography signals using deep learning methods," *Sensors*, vol. 19, no. 7, p. 1736, Apr. 2019.

[35] F. Wang, S. Zhong, J. Peng, J. Jiang, and Y. Liu, "Data augmentation for EEG-based emotion recognition with deep convolutional neural networks," in *Proc. Int. Conf. Multimedia Modeling*, 2018, pp. 82–93.

[36] E. S. Salama, "EEG-based emotion recognition using 3D convolutional neural networks," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 8, pp. 329–337, Jun. 2021.

[37] Y. Li, X.-R. Zhang, B. Zhang, M.-Y. Lei, W.-G. Cui, and Y.-Z. 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.

[38] M. Arslan, M. Guzel, M. Demirci, and S. Ozdemir, "SMOTE and Gaussian noise based sensor data augmentation," in *Proc. 4th Int. Conf. Comput. Sci. Eng. (UBMK)*, Sep. 2019, pp. 1–5.

[39] X. Cui, V. Goel, and B. Kingsbury, "Data augmentation for deep neural network acoustic modeling," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 9, pp. 1469–1477, Sep. 2015.

[40] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local Nash equilibrium," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst. (NIPS)*, 2017, pp. 6629–6640.

[41] J. Deng, Z. Zhang, F. Eyben, and B. Schuller, "Autoencoder-based unsupervised domain adaptation for speech emotion recognition," *IEEE Signal Process. Lett.*, vol. 21, no. 9, pp. 1068–1072, Sep. 2014.

[42] N. K. N. Aznan, A. Atapour-Abarghouei, S. Bonner, J. D. Connolly, N. Al Moubayed, and T. P. Breckon, "Simulating brain signals: Creating synthetic EEG data via neural-based generative models for improved SSVEP classification," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2019, pp. 1–8.

[43] Y. Xu, J. Yang, and M. Sawan, "Multichannel synthetic preictal EEG signals to enhance the prediction of epileptic seizures," *IEEE Trans. Biomed. Eng.*, vol. 69, no. 11, pp. 3516–3525, Nov. 2022.

[44] M. Arjovsky and L. Bottou, "Towards principled methods for training generative adversarial networks," 2017, *arXiv:1701.04862*.

[45] C. Brunner, R. Leeb, G. Muller-Putz, A. Schlogl, and G. Pfurtscheller, "BCI competition 2008—Graz data set A," Inst. Knowl. Discovery, Lab. Brain-Comput. Interfaces, Graz Univ. Technol., Graz, Austria, 2008, pp. 1–6, vol. 16.

[46] C. Tremmel et al., "A meta-learning BCI for estimating decision confidence," *J. Neural Eng.*, vol. 19, no. 4, Jul. 2022, Art. no. 046009.

[47] 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.

[48] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1800–1807.

[49] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 8, pp. 2011–2023, Aug. 2020.

[50] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1520–1528.

[51] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[52] A. Odena, V. Dumoulin, and C. Olah, "Deconvolution and checker-board artifacts," *Distill*, vol. 1, no. 10, p. e3, 2016.

[53] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier GANs," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, Aug. 2017, pp. 2642–2651.

[54] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan, "Unsupervised pixel-level domain adaptation with generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 95–104.

[55] S. Wen et al., "Rapid adaptation of brain-computer interfaces to new neuronal ensembles or participants via generative modelling," *Nature Biomed. Eng.*, Nov. 2021, doi: 10.1038/s41551-021-00811-z.

[56] S. Chatterjee and P. Zielinski, "On the generalization mystery in deep learning," 2022, *arXiv:2203.10036*.

[57] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*.

[58] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local Nash equilibrium," 2017, *arXiv:1706.08500*.

[59] G. Peyré and M. Cuturi, "Computational optimal transport," 2018, *arXiv:1803.00567*.

[60] K. G. Hartmann, R. T. Schirrmeister, and T. Ball, "EEG-GAN: Generative adversarial networks for electroencephalograhic (EEG) brain signals," 2018, *arXiv:1806.01875*.

[61] H. Adeli, Z. Zhou, and N. Dadmehr, "Analysis of EEG records in an epileptic patient using wavelet transform," *J. Neurosci. Methods*, vol. 123, no. 1, pp. 69–87, Feb. 2003.

[62] C. Neuper, M. Wörtz, and G. Pfurtscheller, "ERD/ERS patterns reflecting sensorimotor activation and deactivation," *Progr. Brain Res.*, vol. 159, pp. 211–222, Jan. 2006.

[63] H. Ramoser, J. Müller-Gerking, and G. Pfurtscheller, "Optimal spatial filtering of single trial EEG during imagined hand movement," *IEEE Trans. Rehabil. Eng.*, vol. 8, no. 4, pp. 441–446, Dec. 2000.

[64] Z. Zhang et al., "A novel deep learning approach with data augmentation to classify motor imagery signals," *IEEE Access*, vol. 7, pp. 15945–15954, 2019.

[65] F. Fahimi, S. Dosen, K. K. Ang, N. Mrachacz-Kersting, and C. Guan, "Generative adversarial networks-based data augmentation for brain–computer interface," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 9, pp. 4039–4051, Sep. 2021.

[66] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

[67] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2921–2929.