ORIGINAL ARTICLE

EEG data augmentation for emotion recognition with a multiple generator conditional Wasserstein GAN

Aiming Zhang[1] · Lei Su[1] · Yin Zhang[2] · Yunfa Fu[1] · Liping Wu[1] · Shengjin Liang[1]

Abstract

EEG-based emotion recognition has attracted substantial attention from researchers due to its extensive application prospects, and substantial progress has been made in feature extraction and classification modelling from EEG data. However, insufficient high-quality training data are available for building EEG-based emotion recognition models via machine learning or deep learning methods. The artificial generation of high-quality data is an effective approach for overcoming this problem. In this paper, a multi-generator conditional Wasserstein GAN method is proposed for the generation of high-quality artificial that covers a more comprehensive distribution of real data through the use of various generators. Experimental results demonstrate that the artificial data that are generated by the proposed model can effectively improve the performance of emotion classification models that are based on EEG.

Keywords EEG · Emotion recognition · GAN · Data augmentation

Introduction

Emotion plays an important role in human cognition, namely, in rational decision-making, perception, interpersonal communication, and human intelligence [19]. Positive emotions help improve human health and work efficiency, while negative emotions may cause health problems. The development of devices and systems that can automatically recognize human emotions is of substantial interest to researchers [26, 28].

Emotion recognition has high application prospects in various scenarios and has been widely used in safe driving; medical care, especially mental health monitoring; social security; and other fields [1]. Researchers usually study features that are used to identify human emotions from a variety of perspectives, such as facial expressions, posture, voice, and neurophysiological signals. Then, they extract features from many signals and use machine learning or deep learning methods to identify emotions [23]. For using machine learning or deep learning to build emotion recognition models that are based on EEG, insufficient high-quality training data are available. One of the major bottlenecks in EEG-based emotion recognition is the acquisition of relevant data [17]. This is due to the following main reasons: (a) it is expensive to build an experimental environment that can capture EEG. (b) experiments on EEG-based emotion recognition are time-consuming and tedious, and the efficiency of signal acquisition is low. (c) The signal-to-noise ratio of the original EEG data is too low. (d) Emotion categories are difficult to label accurately. (e) Currently, in EEG-based emotion recognition, the number and volume of public data sets are small. For example, SEED [37], DEAP [13], DREAMER [12], MAHNOB-HCI [27] and eNTERFACE'05 [20], which are common data sets in this field, contain relatively few samples and contain less than 30 subjects' data on average. Therefore, the lack of high-quality EEG data limits the further development and application of machine learning and deep learning methods in EEG-based emotion recognition.

A strategy for solving the problem of data scarcity is to transform the original data to generate artificial data, which is typically referred to as data augmentation [36]. By performing geometric transformations, noise addition, interpolation, and other operations on the original data,

✉ Lei Su
  s28341@hotmail.com

1 School of Information Engineering and Automation, Kunming University of Science and Technology, Kunming, China

2 School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu, China

without obtaining additional real data, the available data can produce values that are equivalent to additional data. A data augmentation method in which data are generated by geometrically transforming the original data has also been successfully applied in the field of EEG-based emotion recognition. Deep learning-based methods provide effective and powerful approaches for learning the implicit expression of data distribution, such as the GAN [7], which can learn how to approach the real distribution of data. In emotion recognition and EEG applications, Hartmann et al. used deep generative models to artificially generate raw EEG data [10]. Yun et al. successfully applied CWGAN for the DE (differential entropy) feature of the SEED dataset, which improved the performance of the classifier [16]. Yun et al. provided a new method for EEG-based emotion recognition and proposed a feasible scheme for measuring the quality of generated data, but the variation in the generated data was not considered further.

Although GAN-based methods can be used to generate realistic data to supplement the real data, since the original EEG signal has a low signal-to-noise ratio, directly generating the original EEG data may introduce noise and artefacts, which would require noise reduction and artefact removals. In addition, in EEG-based emotion recognition tasks, the classifier usually must process the advanced features of EEG data. Morerio et al. and Yun et al. used a GAN to augment data in the data space [16, 31]. In EEG-based emotion recognition tasks, classifiers usually must handle advanced features that have been extracted from raw EEG data, but training GANs in feature space easily leads to mode collapse [6], namely, the generator produces highly similar data, and since the discriminator cannot effectively distinguish the data that are generated by the generator from the real data and, thus, cannot guide the generator to learn the difference between the data, the artificial data that are generated at this time may be too similar. As the amount of artificially generated data increases gradually, more low-quality data will be generated in the artificial data, which will have a negative impact.

In response to the above problems, this paper uses a multiple generator conditional Wasserstein generative adversarial network to implement data augmentation based on EEG. The inclusion of label-based constraints into the model to guide the process of feature generation forces generators to learn various features and to learn the data patterns of real data from various perspectives. Most of the parameters of the generators are shared to reduce the computational burden of the model and share the underlying information. Modification of the gradient penalty term to a zero-centred gradient penalty term enhances the convergence of the model. As the models learn more real data patterns, they are expected to generate artificial feature data with less noise that are closer in distribution to

the real data and retain more diversity in the same type of data.

The remainder of this paper is organized as follows: "Related work" introduces related work. "Multiple generator conditional Wasserstein GAN" introduces the multiple generator conditional Wasserstein GAN that is proposed in this paper. "Experiment and analysis" introduces the experiments and presents the results. Finally, "Conclusions" summarizes the findings.

Related work

EEG-based emotion recognition

The latest development of and research on brain-computer interface (BCI) technology have promoted emotion detection and classification, and EEG-based emotion recognition has attracted widespread attention. The proliferation of portable wireless EEG devices, advanced computational intelligence technologies, and machine learning have accelerated research in this field [2].

Zander et al. introduced affective factors into the traditional BCI [35] and defined the affective brain-computer interface (aBCI) [23]. Alarcao and Fonseca investigated EEG-based emotion recognition methods, starting from the physiological basis of emotion and psychological research, and compared the primary aspects that are involved in the process of emotion recognition, which includes subjects, acquisition equipment, modes of emotion stimulation, methods of feature extraction and classifiers [1]. Abeer et al. systematically reviewed the current status and development trend of EEG-based emotion recognition from two aspects: research methodology and classification methods [2].

In this research, various emotional stimulation methods are used to elicit emotional responses from the subjects. The common methods are the use of music [13], pictures [15] and video clips [37]. Among these stimuli, movie editing is considered to be one of the most effective ways to trigger human emotions. Zheng and Lu recruited 15 subjects to watch 15 selected Chinese movie clips to induce three emotions (SEED data set) [37]. Koelstra et al. developed a public EEG-based emotion dataset, namely, DEAP [13], by recruiting 32 participants to watch 40 music videos.

EEG-based emotion recognition has been applied to two broad fields: medical and non-medical. In the medical field, EEG-based emotion recognition systems are used to assist, monitor, enhance or diagnose the emotional state of patients. Some are based on the automatic classification of normal and depression-related EEG signals, which are used as diagnostic and monitoring tools to detect depression [4]. Maglione et al. discussed an EEG-based emotion detection system for patients with cochlear implants [18]. Other medical cases

that are related to emotions, which include schizophrenia, autism, bipolar disorder, epilepsy, attention-deficit/hyperactivity disorder, bulimia nervosa, borderline personality disorder and other mental illnesses, have also received attention from many researchers [2]. In addition, there are many products and applications in non-medical fields, such as lie detection, stress detection, alertness and attention detection, and detection of emotional feedback from language, pictures, music and videos [2]. Various products can even be purchased in online shopping malls.

Generative adversarial networks

A generative model aims at learning a specified dataset's data distribution through unsupervised learning, thereby generating fresh data with modifications. Generative models have been widely investigated and applied in the field of machine learning, with data that include images [33], text, and speech. The generative adversarial network (GAN) [7] is the most promising and effective generative model. The original GAN is a network that is composed of generators and discriminators. The two parts of the network compete and promote each other and eventually reach Nash equilibrium. The generator learns how to generate artificial data that are similar to the real data, while the discriminator evaluates the probability that a sample originates from the real data or artificial data. In the training of a GAN, the generator attempts to deceive the discriminator by generating realistic data, while the discriminator attempts to improve its discrimination performance to avoid being confused by artificial data. After a period of confrontational training, the generator can generate high-quality artificial data.

Mirza and Osindero et al. realized control over the generated data types by adding conditional information [22]. GANs also demonstrate the prospect of generating realistic data in various fields, and increasingly many researchers use GANs to supplement sample data [5]. GANs have been successfully applied to EEG-related studies. For example, the EEG-GAN, which was proposed by Hartmann et al., could be used to generate raw EEG signals [10]. GANs have been used for the detection and diagnosis of brain diseases [34] and for the conversion of raw EEG signals into pictures [24], among other applications.

Although GANs show strong generative performance, many problems are encountered in the process of GAN training, which includes poor convergence, mode collapse, and gradient disappearance. The training instability (non-convergence) that is caused by adversarial training is the most important problem of the original GAN. Researchers have changed the structure of the original GAN or the loss function to increase the stability of training. Arjovsky et al. used the Wasserstein distance to replace the loss function of the original GAN, which significantly increased the training

stability of the GAN while maintaining the generating performance of the GAN[3]. Moreover, WGANs can be easily used for data generation in various fields without professional transformations or special network structures [16].

Data augmentation methods with EEG

The objective of data augmentation is to generate fresh data for a provided data set by transforming the original data while retaining the label information [30]. Since the artificial data that are generated by GANs are similar to the original data in terms of data distribution and artificial data could be used to increase the number of training data to alleviate the lack of training data [32], this method is usually used to reduce the overfitting of the model and improve the performance of the classifier [14]. Data augmentation technology has been widely used in computer vision, natural language processing, speech processing, and other fields.

In a study of Zheng et al., DCGAN was used to generate images, and the images were added to the training set to increase the performance of identity recognition tasks [38]. Their results proved the feasibility of data enhancement based on a GAN, and the anti-overfitting ability of the classifier could be enhanced by adding the generated training samples. In terms of EEG data augmentation, EEG-GAN, which was proposed by Hartmann et al., could generate raw EEG data and established a new line of investigation [10]. In addition, they proposed evaluation indicators for evaluating the EEG data that are generated by GANs. However, they did not discuss the performance gain of the classifier. Yun et al. extended the GAN-based augmentation method to EEG-based emotion recognition [16], and the experimental results demonstrated that the GAN-based data augmentation method was effective for EEG-based emotion recognition.

Multiple generator conditional Wasserstein GAN

WGAN

GANs consist of two competing components which are both parameterized as deep neural networks [17]. Given noise (prior distribution) as input, a generator G generates artificial data, and a discriminator D attempts to distinguish whether the sample originates from the real data distribution X_r or the generated data distribution X_g. The generator is optimized to generate realistic data to confuse the discriminator. Both parts of the network are optimized simultaneously to reach the Nash equilibrium state. The objective function of the model is as follows [7]:

$$\min_{\theta_G}\max_{\theta_D} L(X_r, X_g) = \mathbb{E}_{x_r \sim X_r}[\log(D(x_r))] + \mathbb{E}_{x_z \sim Z}[\log(1 - D(G(z)))]$$
$$= \mathbb{E}_{x_r \sim X_r}[\log(D(x_r))] + \mathbb{E}_{x_g \sim X_g}[\log(1 - D(G(z)))]\# \tag{1}$$

where θ_G and θ_D represent the parameters of the generator and the discriminator, respectively, and z can be uniform noise or Gaussian noise.

Neither the Kullback–Leibler (KL) divergence nor the Jensen–Shannon (JS) divergence can provide sufficient gradients for the generator, which is the main reason for GAN training instability and gradient disappearance [3]. At the beginning of the training, the discriminator easily determines whether the received data are real data; hence, the discriminator in the GAN is easily trained as the (approximate) optimal discriminator. When the (approximate) optimal discriminator is trained, minimizing the loss of the generator is equivalent to minimizing the JS divergence between X_r and X_g, and since it is almost impossible for X_r and X_g to have a non-negligible overlap, regardless of their distance, the JS divergence is constant log 2, which causes the gradient of the generator to be (approximately) 0, which, in turn, causes the gradient to disappear.

Arjovsky et al. used the Wasserstein distance (which is also known as EMD) to measure the distance between two distributions, which is a substantial contribution to resolving the instability of GAN training and the gradient disappearance [3]. The Wasserstein distance is defined as follows:

$$W(X_r, X_g) = \inf_{\gamma \sim \Pi(X_r, X_g)} \mathbb{E}_{(x_r, x_g) \sim \gamma}[x_r - x_g], \tag{2}$$

where $\Pi(X_r, X_g)$ represents the collection of the joint probability distribution between X_r and X_g. Even if there is no overlap between the two distributions, the Wasserstein distance can still provide a useful and smooth gradient for GAN training. However, $\inf_{\gamma \sim \Pi(X_r, X_g)}$ in the Wasserstein distance definition (formula (2)) cannot be solved directly. One strategy for calculating the Wasserstein distance is to use its Kantorovich–Rubinstein duality:

$$W(X_r, X_g) = \frac{1}{K} \sup_{\|f\|_L \leq K} \mathbb{E}_{x_r \sim X_r}[f(x_r)] - \mathbb{E}_{x_g \sim X_g}[f(x_g)], \tag{3}$$

where f is the set of 1-Lipschitz functions and K is a constant. In the implementation, the discriminator D replaces f, and $\|f\|_L \leq K$ is replaced by $\|f\|_L \leq K$.

Many methods can be used to enforce the 1-Lipschitz constraints in the implementation of WGAN. One approach to limit the parameters of the discriminator to a specified range, for example, $-0.1 \sim 0.1$. However, this cutting method will result in various problems. Since clipping reduces the capacity of the model, the model may generate low-quality data and have difficulty converging.

Another solution, which was proposed by Arjovsky et al., is to use a gradient penalty and add a penalty term to the loss function [9]:

$$\min_{\theta_G}\max_{\theta_D} L(X_r, X_g) = \mathbb{E}_{x_r \sim X_r}[D(x_r)] - \mathbb{E}_{x_g \sim X_g}[D(x_g)]$$
$$- \lambda \mathbb{E}_{\hat{x} \sim \hat{X}}\left[(\nabla_{\hat{x}} D(\hat{x})_2 - 1)^2\right], \tag{4}$$

λ in formula (4) is a hyperparameter that controls the trade-off between the original target and the gradient penalty. $\hat{x}$ denotes that the real data distribution X_r and the generated data distribution X_g are sampled once and, subsequently, another random sampling is conducted on the line that connects the two points:

$$\hat{x} = \alpha x_r + (1 - \alpha)x_g, \alpha \sim U[0,1], x_r \sim X_r, x_g \sim X_g. \tag{5}$$

Multiple generator conditional Wasserstein GAN (MG-CWGAN)

Yun et al. proposed CWGAN and applied it to EEG-based emotion recognition signals, which generated emotion characteristic data of specified categories [16]. The formula for CWGAN is as follows:

$$\max_{\theta_D} L(X_r, X_g, Y_r) = \mathbb{E}_{x_r \sim X_r, y_r \sim Y_r}[D(x_r y_r)] - \mathbb{E}_{x_g \sim X_g, y_r \sim Y_r}[D(x_g y_r)]$$
$$- \lambda \mathbb{E}_{\hat{x} \sim \hat{X}, y_r \sim Y_r}\left[(\nabla_{\hat{x}y_r} D(\hat{x}y_r)_2 - 1)^2\right], \tag{6}$$

$$\min_{\theta_G} L(X_g, Y_r) = -\mathbb{E}_{x_g \sim X_g, y_r \sim Y_r}[D(x_g \mid y_r)]. \tag{7}$$

In formulas (6) and (7), $Y_r Y_r$ represents the category of the real data, and $\hat{x}$ is defined in formula (5). In their study, λ is set to 10. The last term in formulas (6) indicates that if the gradient norm deviates from its target norm, a penalty is imposed on the model.

Training GANs in feature space is challenging, as they can easily become mired in the problem of mode collapse [6]. The generator generates samples with features that are concentrated in the real data, but the discriminator is unable to identify the differences between these features. An effective method is to use the mixture distribution that is generated by the mixture generator to approximate the feature distribution [11] and encourage the generators to learn the feature patterns of various distributions. However, the direct addition of multiple generators will harm the convergence of the model because many generators are computationally expensive to train continuously and there is no mechanism for forcing each generator to learn different features or enforcing divergence. Therefore, we introduced label-based conditional constraints to guide the

feature generation process. The discriminator learns real data based on label constraints and further forces generators to learn specified features.

The proposed model is illustrated in Fig. 1. The generator is composed of input layers and parameter sharing layers. It accepts a joint input of a prior distribution with guidance labels, while the discriminator accepts a joint input of real data with labels, along with artificial data with guidance labels. The discriminator returns gradients to guide the generator to learn features of the real data. The formula for MG-CWGAN is as follows:

$$\max_{\theta_D} L(X_r, X_g, Y_r) = \mathbb{E}_{x_r \sim X_r, y_r \sim Y_r}[D(x_r y_r)] - \mathbb{E}_{x_g \sim X_g, y_r \sim Y_r}[D(x_g y_r)]$$
$$- \lambda \mathbb{E}_{\hat{x} \sim \hat{X}, y_r \sim Y_r}\left[\left(\nabla_{\hat{x} y_r} D(\hat{x} y_r)_2 - 0\right)^2\right], \quad (8)$$

$$\min_{\theta_{G_{1:N}}} L(X_g, Y_r) = \sum_i^N -\mathbb{E}_{x_g \sim X_g, y_r \sim Y_r}[D(x_{g_i} \mid y_r)], \quad (9)$$

where θ_D and $\theta_{G_{1:N}}$ represent the parameters of the discriminator and number generators, respectively; X_g represents the distribution of the generated data; X_r represents the distribution of the real data; Y_r represents the category distribution of the real data; and $\hat{x}$ is defined in formula (5). For generator $G_{1:N}$, the noise vector Z that is sampled from P_z and the label information Y are connected as $(z \mid y_r)$ as input and sent to different generators to generate a specified type of y feature x.

Formula (8) is obtained by modifying the gradient penalty in (6) according to the suggestions of Lars et al. [21]. On the basis of formula (7), formula (9) uses various types of data to train a discriminator and several generators in generator group G and forces the generator to learn the data patterns of various types of data. To increase the training efficiency of the model, a similar method to that in [32] was adopted for sharing all the parameters of the generator $G_{1:N}$ except those of the input layer. During the training process of the model, the model training expenses were minimized while sharing the real data background information.

Experiment and analysis

First, we evaluate the learning efficiency and quality of the CWGAN model and the MG-CWGAN model under the same conditions. Second, we discuss whether the MMD distance can be a satisfactory measure of the quality of the generated data. Then, we apply various classifiers in semi-supervised training to evaluate the performance of the generated method. Finally, the generated data are visualized and compared with the original data.

Data set

The SEED dataset [37] contains EEG signals from 15 participants who watched 15 edited video clips, which could arouse three types of emotions: positive, neutral, and negative. Each participant participated in 3 experiments at least 7 days apart, for a total of 45 experiments, in which an ESI NeuroScan system was used to sample the signal from a 62-electrode headset at a frequency of 1000 Hz.

In this article, the first experiment in the SEED data set, in which each participant participated, was selected. The number of labelled samples per subject was 3394. The subjects in the original data set were divided into an original training set and an original verification set at a ratio of 2010:1384. The original training set was divided into a training set and a test set after fivefold cross-validation. The DE feature augmented data set that was processed by LDA in the SEED data set was selected.

Evaluation method

The quality of the generated EEG data was evaluated in terms of the MMD distance and the Wasserstein distance (discriminator loss) and via the TSNE visualization

Fig. 1 Illustration of the proposed MG-CWGAN
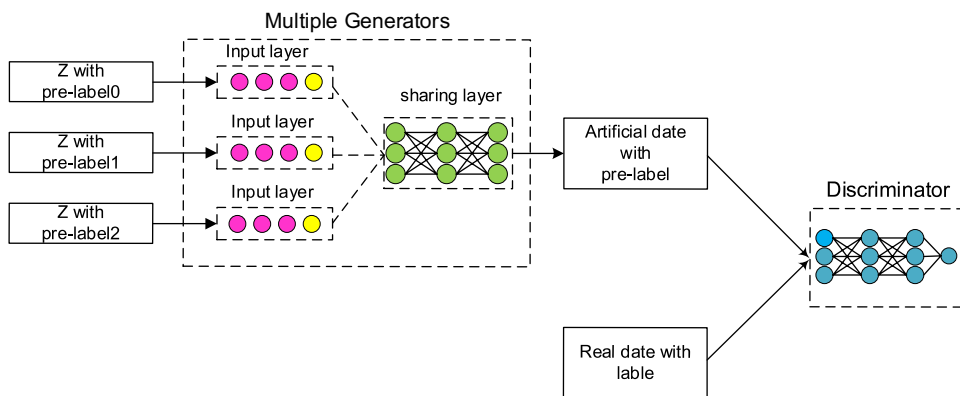
method. Finally, a semi-supervised self-trained classifier was used to evaluate the average recognition accuracy of the data as the experimental accuracy.

1. Wasserstein distance [3] (discriminator loss): it represents the earth-mover distance (EMD) between the real data X_r and the generated data X_g when the network converges. EMD is the minimum cost that is required to transform one distribution into another. Similar to the KL divergence, it can be used to characterize the similarity between two distributions.

2. Maximum mean discrepancy (MMD): MMD [8] is often used to measure the distance between two distributions.

3. t-distributed stochastic neighbor embedding (TSNE) [29]: it is a machine learning method for dimensionality reduction that can map high-level data to a two-dimensional space while maintaining the local structure of the data.

4. Semi-supervised self-training: in semi-supervised self-training, the labelled initial data set (X_{train}, Y_{train}) is used for training to obtain the initial classifier C_{int}, and C_{int} is used to classify the unlabelled data. The "pseudo-labelled" data set (X_g, Y_c), the "pseudo-labelled" data set (X_g, Y_c) and the labelled initial training set (X_{train}, Y_{train}) are obtained, which are used to train the classifier C together, Finally, the performance of classifier C is validated on the initial validation set. However, some "pseudo-marked" data will definitely be incorrect during this process. When sufficiently many "pseudo-labels" are incorrect, the self-training algorithm will reinforce incorrect classification decisions, thereby degrading the performance of the classifier. Semi-supervised classifiers can be used to distinguish the quality of the generated data.

Hyperparameter and network structure selection

In this section, experiments are conducted to evaluate the influence of various parameters and network structures on the model. A grid search is performed on the learning rate, the number of network layers and the batch size of the deep neural network classifier. The Adam optimizer is used to select the learning rate from 0.0005, 0.0001, 0.0005, and 0.00001. The number of hidden layers is searched from 2 to 5. The batch size is selected from 128, 256, and 512. Noise is sampled from the uniform distribution U [− 1,1]. For the parameter group, 10,000 iterations are conducted to fully compare the effects of the parameters. In addition, the input data is compared to determine whether it was shuffled. For the SEED dataset, the dimension of the label is 3, and the dimension of the data feature is 310. The ReLU activation function is used for all hidden layers. Before feeding DE features into the network, the data must be standardized. The experiment is conducted with the same network settings except for the variables (Fig. 2).

The relatively optimal learning rate is around the interval [0.00005, 0.0001], and the fluctuation at approximately 0.00005 is smaller. The deeper the network is, the slower the training and the stronger the instability. MG-CWGAN outperforms CWGAN overall at various learning rates (Figs. 3, 4).

The number of hidden layers has minimal effect on the MG-CWGAN model, whereas CWGAN is affected more strongly. CWGAN with only three hidden layers is unstable in the later stage (Fig. 5).

In an experiment on MG-CWGAN, it is clearly observed that the training speed with a batch size of 128 is the slowest and the stability is low, whereas satisfactory convergence with low divergence is realized with a batch size of 256, and fast and satisfactory convergence is realized with a batch size of 512 (Fig. 6).
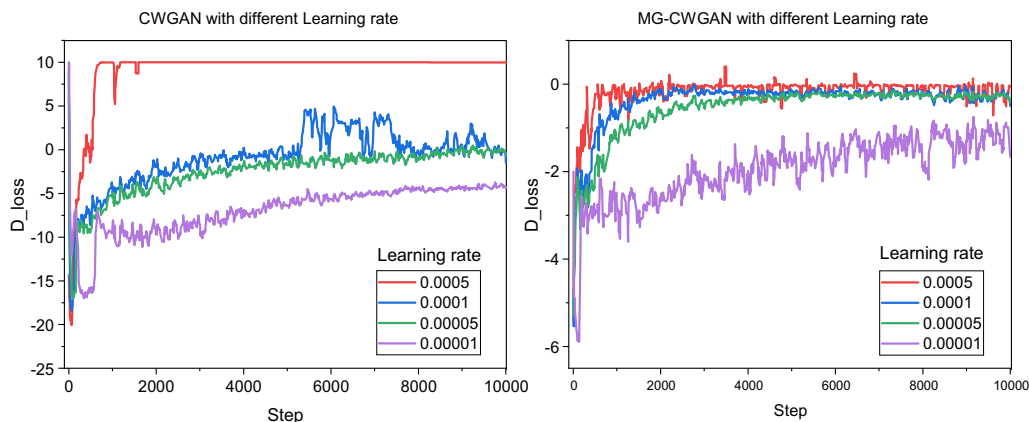

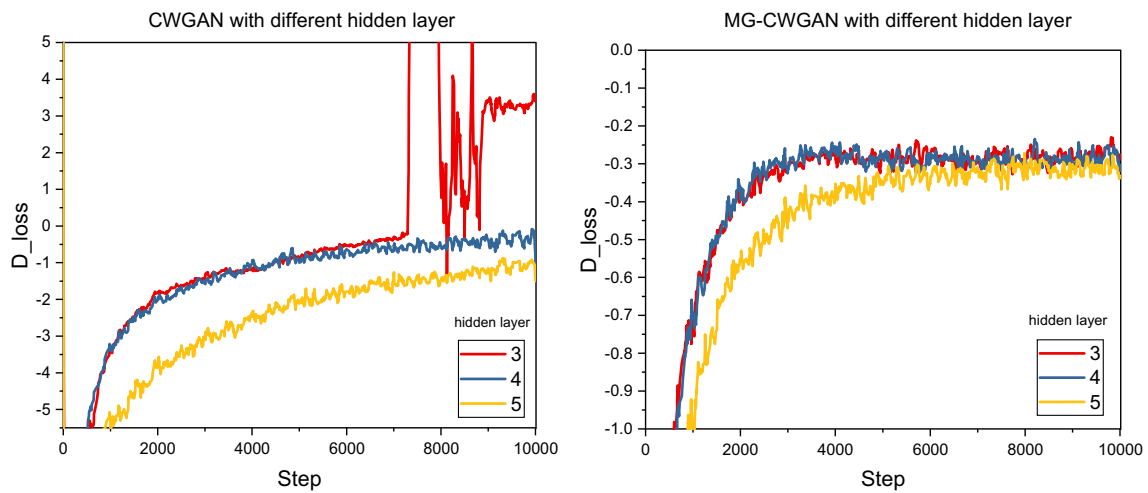**Fig. 2** Impact of the number of hidden layers on D_loss(discriminator loss)

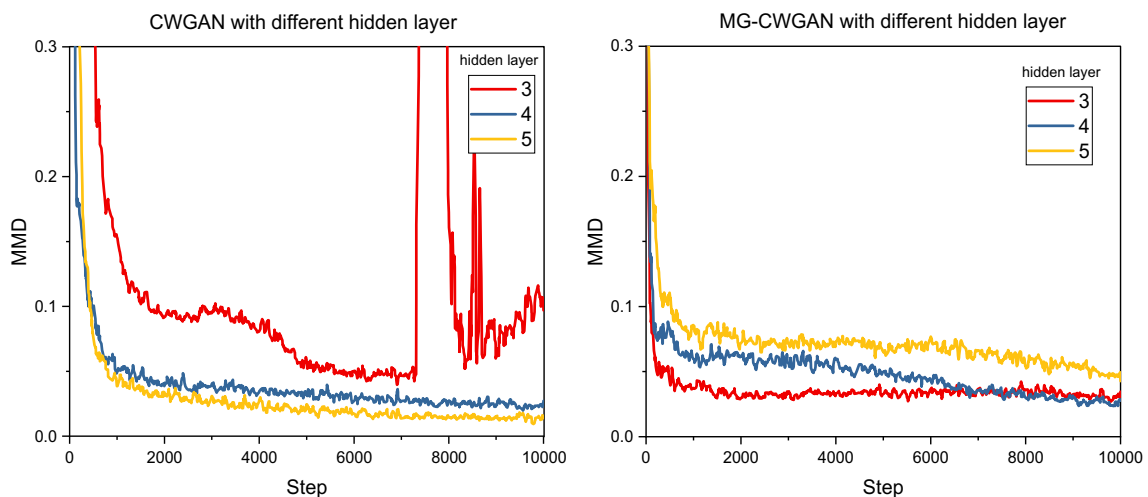**Fig. 3** Impact of the number of hidden layers on *D*_loss

Fig. 4 Impact of the number of hidden layers on MMD

Simply adding generators for various types of signals slightly increases the stability of the network, but substantially reduces the convergence speed of the model; hence, the gradient penalty is changed to address this problem. Inspired by a study of Lars et al. [21], the one-centered gradient penalties are modified to zero-centered gradient penalties, which substantially strengthens the network convergence performance (Fig. 7).

Comparing the discriminator loss (Wasserstein distance) of CWGAN and MG-CWGAN under the same parameters, the overall convergence speed of MG-CWGAN is substantially increased compared with that of CWGAN, and the Wasserstein distance after MG-CWGAN is stabilized fluctuates within the range of [0.2, 0.4], while that of CWGAN fluctuates within the range of [0, 1] and even diverges. Thus,

the MG-CWGAN model realizes larger improvements in convergence speed and stability than the CWGAN model.

MMD applicability

In this section, experiments are conducted to determine whether MMD can evaluate the quality of the generated data well. In this paper, interval sampling (strategy A) and random data scrambling (strategy B) are used for data sampling. After sampling, the data are grouped, and the MMD distance between the groups are calculated. The MMD distance can prove that the model can shorten the distance between the real data and the generated data, but it may not be suitable for evaluating the quality of the generated EEG data alone.
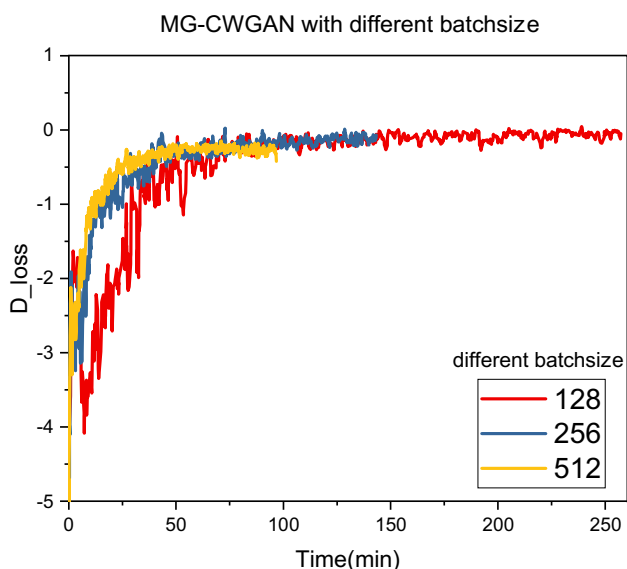
Fig. 5 Impact of the batch size

Experimental results demonstrate that its sensitivity to the number of data points exceeds that between categories.

MMD distance between subjects

Among subjects, strategy A was used to extract 675 data items from each subject's first experimental data to calculate the MMD distance between pairs of subjects. The mean MMD distance between each subject and other subjects was 4.237, and the overall standard deviation was 0.94. The minimum value was 1.74, and the maximum value was 6.44.

MMD distance within subjects

For the same subject, strategy A is used to calculate the MMD distance in three scenarios: (1) the data are evenly distributed among 5 groups in order (675 data in each group), and the MMD distance between groups is calculated. The mean is 2.16736 and the standard deviation is 0.57. (2) The MMD distance is calculated between the same emotions of the selected subjects (185 data in each group). The mean and standard deviation of MMD between positive emotions are 4.79447 and 0.82729, respectively, and between neutral emotions, the mean and standard deviation of the MMD are 4.96198 and 0.89172, respectively, whereas the mean and standard deviation of the MMD between negative emotions are 5.62587 and 0.72508, respectively. (3) For the MMD distance between the subjects' emotions (185 data in each group), the mean and the standard deviation are 5.1502 and 0.57701, respectively (Table 1).

For the same subject, strategy B is adopted, and this case is divided into three scenarios to calculate the MMD distance: (1) The data are divided into 5 groups (675 data points in each group), and the MMD distance between them is calculated. The mean and standard deviation are 0.0077 and 0.003, respectively. (2) The data are divided into 10 groups (330 data points in each group), and the MMD distance between them is calculated. The mean and standard deviation are 0.01824 and 0.00613, respectively. (3) The data are divided into 15 groups (200 data points in each group), and the MMD distance between them is calculated. The mean and standard deviation are 0.0262 and 0.013, respectively. As the number of data in the group decrease, the MMD distance gradually increases (Table 2).
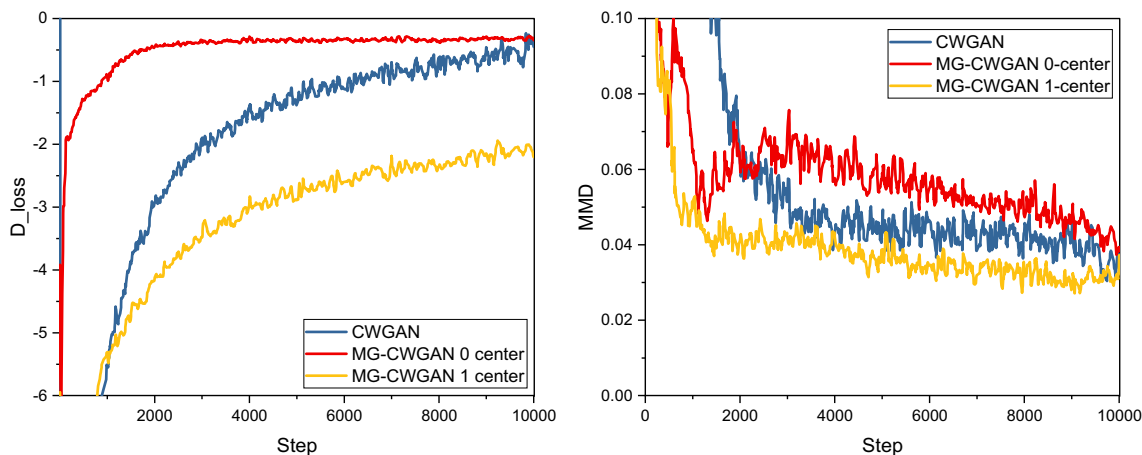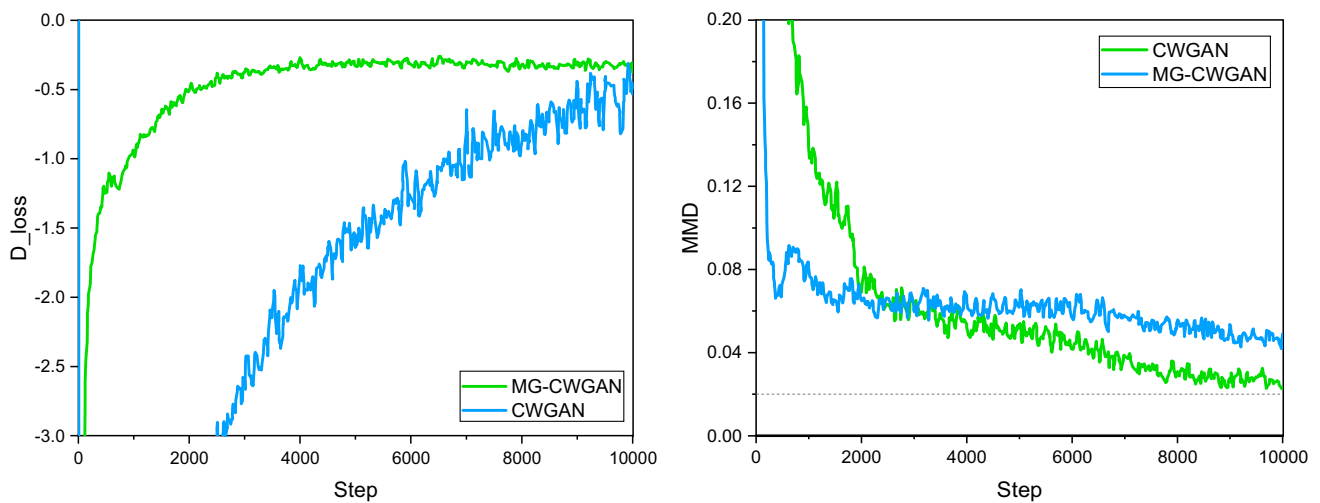

Fig. 6 Impact of the gradient penalty

Fig. 7 Comparison with the same parameters

Table 1 Strategy A for the same subject in scenario 1

Group	1	2	3	4	5	Mean/std
1	N/A	1.4256	2.4507	2.4855	2.6041	2.2415/0.47
2	1.4256	N/A	2.5744	2.5297	2.8039	2.3334/0.53
3	2.4507	2.5744	/N/A	0.9387	2.1391	2.0257/0.64
4	2.4855	2.5297	0.9387	N/A	1.7219	1.9189/0.65
5	2.6041	2.8039	2.1391	1.7219	N/A	2.3173/0.41
	2.2415/0.47	2.3334/0.53	2.0257/0.64	1.9189/0.65	2.3173/0.41	2.1674/0.57

Table 2 Strategy B for the same subject in scenario 1

Group	1	2	3	4	5	Mean/std
1	N/A	0.0068	0.0154	0.0059	0.0097	0.0095/0.004
2	0.0068	N/A	0.0100	0.0055	0.0077	0.0075/0.002
3	0.0154	0.0100	N/A	0.0071	0.0050	0.0059/0.003
4	0.0059	0.0055	0.0071	N/A	0.0034	0.0055/0.001
5	0.0097	0.0077	0.0050	0.0034	N/A	0.0065/0.002
	0.0095/0.004	0.0075/0.002	0.0059/0.003	0.0055/0.001	0.0065/0.002	0.0077/0.003

Performance of the classifier

In this paper, the DE features that are processed via LDA in the SEED dataset are selected to train the network and generate artificial data to expand the dataset. In each experiment, 3394 samples of all trials of a subject are selected. The first 2010 data are used as the original training set, and the last 1384 data are used as the original validation set. Artificial samples of sizes 0, 300, 600, 900, 1500, 3000, 6000, 9000, and 15,000 are generated for DE features and added to the original training set as the training set of the classifier. SVM and KNN are used as the classifiers. These artificial data are added to the original training set to form a new training set, the classifiers are retrained on the new

training set, and finally, the classification effect is evaluated on the original validation set. Here, "0" denotes that only the actual DE data are being used, without the addition of any human data.

The colours represent the amounts of additional data. 0 represents negative emotions, 1 represents neutral emotions, and 2 represents positive emotions.

The classification performances of SVM and KNN were evaluated under the addition of various amounts of generated EEG data. After semi-supervised training by SVM, as shown in Fig. 8, the data that are generated by MG-CWGAN are beneficial to the algorithm in most cases. The classification performance is improved, but the classification accuracy of the algorithm for the data that were generated by CWGAN
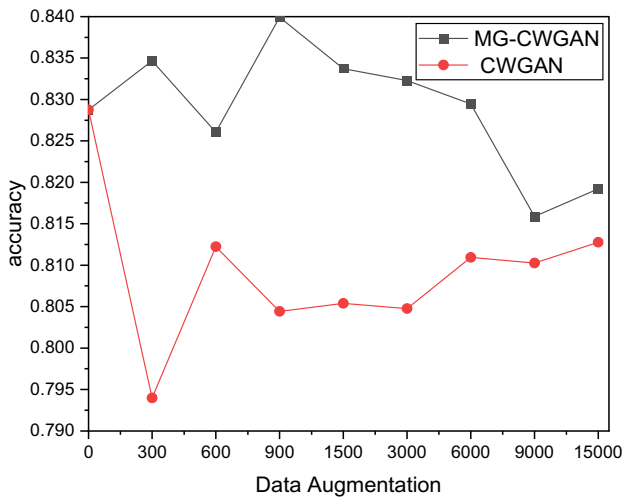
Fig. 8 Performance of SVM

has decreased; hence, the data that were generated by MG-CWGAN are closer to the original data. Figure 9 shows the impacts of the data that were generated by MG-CWGAN on the precision and recall of the classifier (Fig. 10).

Figure 8 shows that the artificial data that were generated by the two methods can play a role in increasing the classification accuracy of the KNN classifier. Since KNN is highly robust to noise in the training data, it is also highly effective when given a sufficiently large training set, and it is not sensitive to a small number of outliers; in practice, KNN uses all the attributes of the instance (features) [25] to calculate the distance. Nonetheless, the distance between neighbours will be dominated by many irrelevant attributes, which will negatively affect the classification accuracy.

Visualization of the generated data

In this section, the data are visualized via the t-SNE [29] method, which plots the distributions of the real and generated DE features. The data that were generated by CWGAN and the data that were generated by MG-CWGAN are compared to a manifold in two-dimensional space. A two-dimensional visualization of the emotions of the same subject in the latent space by the t-SNE method is shown in Fig. 11. A two-dimensional visualization of the generated data and real data in the latent space by the t-SNE method is shown in Fig. 12, according to which the generated data carry sufficient real information. The distribution of the generated data is similar to that of the real data. The red, yellow and purple data points represent negative, neutral and positive emotions respectively.

In Fig. 12a–c are generated by CWGAN, and d to f are generated by MG-CWGAN. The red, yellow, and purple data points represent negative, neutral, and positive emotions, respectively, and the blue data points represent generated data. The light blue circular area marks the low-quality area.

After visualization by t-SNE, the distributions of the data that were generated by the two methods and the real data are compared. The data that were generated by CWGAN are mixed together; hence, low-quality artificial data were produced. From the data that were generated by MG-CWGAN, more features of the real data were learned, and the generated data are close to the corresponding real data; hence, the generated data carry more information about the real data, and there is less information when generating more data. The distributions of the generated data are mixed with each other.
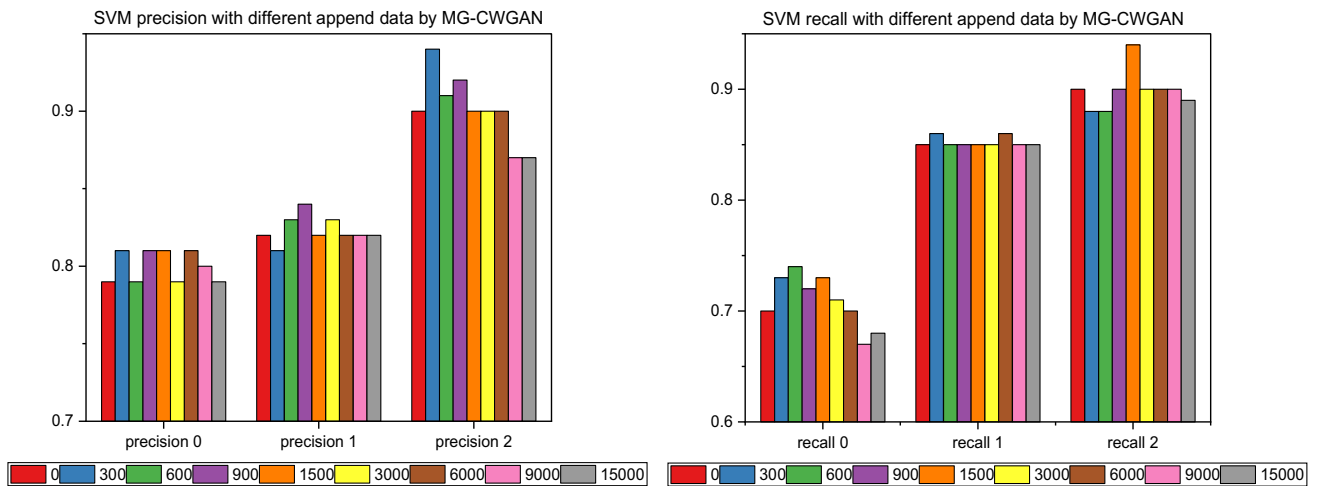

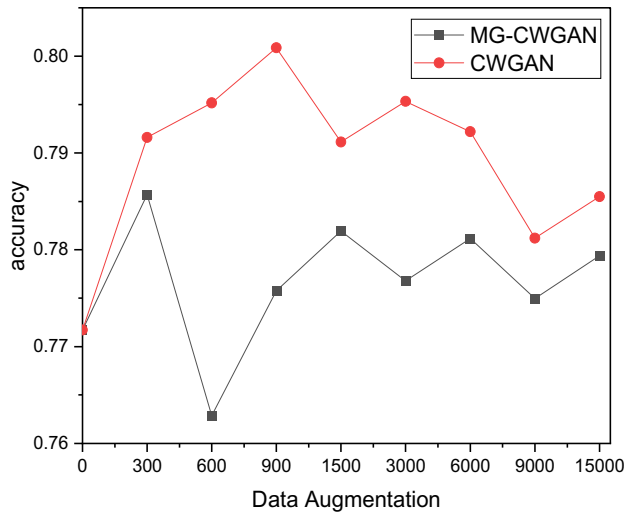**Fig. 9** Impacts of generated data on the precision and recall rates of various emotions

Fig. 10 Performance of KNN

Fig. 11 t-SNE visualization of feature distributions of one subject in the dataset

Conclusions

This paper proposes a multi generator conditionally Wasserstein GAN model for EEG data augmentation to enhance EEG-based emotion recognition. This paper uses the generated data to expand the original training data set and evaluates the quality of the generated data and the accuracy of the EEG-based emotion recognition model by semi-supervised self-training. The experimental results on two classifiers prove the satisfactory performance of this method. The generated data are visualized, and the reasons for the increased accuracy are discussed.

We believe that the proposed method can better supplement the experimental data of relevant studies. However, different from images, sounds and texts that can be calculated manually, the generated data are too abstract, and how to evaluate the quality of these generated artificial data needs further research. Additionally, allowing the model to simultaneously learn the emotional data of different subjects so that more general data characteristics may be learned should be considered.
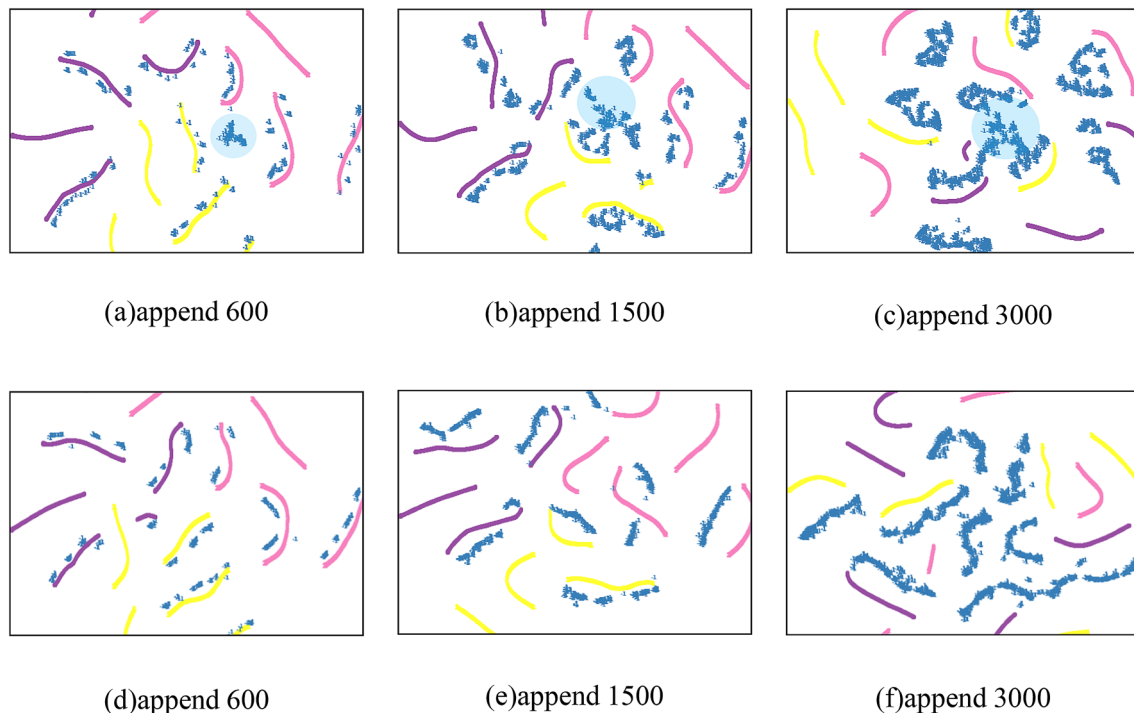
Fig. 12 t-SNE visualization of the real and generated DE feature distributions of one subject in the SEED dataset

References

1. Alarcao SM, Fonseca MJ (2017) Emotions recognition using EEG signals: a survey. IEEE Trans Affect Comput 10:374–393
2. Al-Nafjan A, Hosny M, Al-Ohali Y, Al-Wabil A (2017) Review and classification of emotion recognition based on EEG brain–computer interface system research: a systematic review. Appl Sci 7:1239
3. Arjovsky M, Chintala S, Bottou L (2017) Wasserstein generative adversarial networks. In: International conference on machine learning, 2017. PMLR, pp 214–223
4. Bairy GM, Niranjan U, Puthankattil SD (2016) Automated classification of depression EEG signals using wavelet entropies and energies. J Mech Med Biol 16:1650035
5. Douzas G, Bacao F (2018) Effective data generation for imbalanced learning using conditional generative adversarial networks. Expert Syst Appl 91:464–471
6. Goodfellow I (2016) Nips 2016 tutorial: generative adversarial networks. arXiv preprint arXiv: 170100160
7. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. In: Advances in neural information processing systems, vol 27. pp 2672–2680
8. Gretton A, Borgwardt K, Rasch M, Schölkopf B, Smola A (2006) A kernel method for the two-sample-problem. Adv Neural Inf Process Syst 19:513–520
9. Gulrajani I, Ahmed F, Arjovsky M, Dumoulin V, Courville A (2017) Improved training of wasserstein gans. arXiv preprint arXiv: 170400028
10. Hartmann KG, Schirrmeister RT, Ball T (2018) EEG-GAN: generative adversarial networks for electroencephalograhic (EEG) brain signals. arXiv preprint arXiv: 180601875
11. Hoang Q, Nguyen TD, Le T, Phung D (2018) MGAN: training generative adversarial nets with multiple generators. In: International conference on learning representations, 2018
12. Katsigiannis S, Ramzan N (2017) DREAMER: a database for emotion recognition through EEG and ECG signals from wireless low-cost off-the-shelf devices. IEEE J Biomed Health Inform 22:98–107
13. Koelstra S, Muhl C, Soleymani M, Lee J-S, Yazdani A, Ebrahimi T, Pun T, Nijholt A, Patras I (2011) Deap: a database for emotion analysis; using physiological signals. IEEE Trans Affect Comput 3:18–31
14. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. Adv Neural Inf Process Syst 25:1097–1105

15. Lang PJ, Bradley MM, Cuthbert BN (1997) International affective picture system (IAPS): technical manual and affective ratings. NIMH Center Study Emot Atten 1:39–58

16. Luo Y, Lu B-L (2018) EEG data augmentation for emotion recognition using a conditional Wasserstein GAN. In: 2018 40th annual international conference of the IEEE engineering in medicine and biology society (EMBC), 2018. IEEE, pp 2535–2538

17. Luo Y, Zhu L-Z, Wan Z-Y, Lu B-L (2020) Data augmentation for enhancing EEG-based emotion recognition with deep generative models. J Neural Eng 17:056021

18. Maglione A, Scorpecci A, Malerba P, Marsella P, Giannantonio S, Colosimo A, Babiloni F, Vecchiato G (2015) Alpha EEG frontal asymmetries during audiovisual perception in cochlear implant users. Methods Inf Med 54:500–504

19. Marg E (1995) DESCARTES'ERROR: emotion, reason, and the human brain. Optom Vis Sci 72:847–848

20. Martin O, Kotsia I, Macq B, Pitas I (2006) The eNTERFACE'05 audio-visual emotion database. In: 22nd international conference on data engineering workshops (ICDEW'06), 2006. IEEE, pp 8–8

21. Mescheder L, Geiger A, Nowozin S (2018) Which training methods for GANs do actually converge? In: International conference on machine learning, 2018. PMLR, pp 3481–3490

22. Mirza M, Osindero S (2014) Conditional generative adversarial nets. arXiv preprint arXiv: 14111784

23. Mühl C, Allison B, Nijholt A, Chanel G (2014) A survey of affective brain computer interfaces: principles, state-of-the-art, and challenges. Brain Comput Interfaces 1:66–84

24. Palazzo S, Spampinato C, Kavasidis I, Giordano D, Shah M (2017) Generative adversarial networks conditioned by brain signals. In: Proceedings of the IEEE international conference on computer vision, 2017, pp 3410–3418

25. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V (2011) Scikit-learn: machine learning in Python. J Mach Learn Res 12:2825–2830

26. Smith D, Burke B (2019) Hype cycle for emerging technologies. Gartner, Inc Abgerufen am 12:2019

27. Soleymani M, Lichtenauer J, Pun T, Pantic M (2011) A multimodal database for affect recognition and implicit tagging. IEEE Trans Affect Comput 3:42–55

28. Somers M (2019) Emotion AI, explained. Режим доступа: https://mitsloanmit.edu/ideas-made-to-matter/emotion-ai-explained. Дата обращения 20

29. Van der Maaten L, Hinton G (2008) Visualizing data using t-SNE. J Mach Learn Res 9

30. Van Dyk DA, Meng X-L (2001) The art of data augmentation. J Comput Graph Stat 10:1–50

31. Volpi R, Morerio P, Savarese S, Murino V (2018) Adversarial feature augmentation for unsupervised domain adaptation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018. pp 5495–5504

32. Wang S-H, Govindaraj VV, Górriz JM, Zhang X, Zhang Y-D (2021) Covid-19 classification by FGCNet with deep feature fusion from graph convolutional network and convolutional neural network. Inf Fusion 67:208–229

33. Wang S-H, Nayak DR, Guttery DS, Zhang X, Zhang Y-D (2021) COVID-19 classification by CCSHNet with deep fusion using transfer learning and discriminant correlation analysis. Inf Fusion 68:131–148

34. Wei Z, Zou J, Zhang J, Xu J (2019) Automatic epileptic EEG detection using convolutional neural network with improvements in time-domain. Biomed Signal Process Control 53:101551

35. Zander TO, Jatzev S (2011) Context-aware brain–computer interfaces: exploring the information space of user, technical system and environment. J Neural Eng 9:016003

36. Zhang Y-D, Dong Z, Chen X, Jia W, Du S, Muhammad K, Wang S-H (2019) Image-based fruit category classification by 13-layer deep convolutional neural network and data augmentation. Multimed Tools Appl 78:3613–3632

37. Zheng W-L, Lu B-L (2015) Investigating critical frequency bands and channels for EEG-based emotion recognition with deep neural networks. IEEE Trans Auton Ment Dev 7:162–175

38. Zheng Z, Zheng L, Yang Y (2017) Unlabeled samples generated by gan improve the person re-identification baseline in vitro. In: Proceedings of the IEEE international conference on computer vision, 2017. pp 3754–3762

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations